

Modelo predictivo para optimizar la calidad del servicio en PayJoy INC.

Maria Fernanda Amaya Mora

Luis Alfredo Morantes Pita

Asesor

Andrés Felipe Hernández Giraldo

Universidad Nacional Abierta y a Distancia UNAD

Escuela de Ciencias Básicas, Tecnología e Ingeniería ECBTI

Especialización en Ciencia de Datos y Analítica

2025

Dedicatoria

Dedicamos este trabajo a nuestras familias, quienes han sido un pilar fundamental durante este proceso. A la pareja de mi compañero, por su apoyo incondicional. En lo personal, a mi madre, por su amor constante y sus palabras de ánimo, y a mi hijo, quien me inspira cada día a seguir adelante con valentía y determinación.

Agradecimientos

Agradecemos a la empresa PayJoy Inc por su generosidad al financiar nuestra formación académica y profesional. Extendemos nuestra gratitud al tutor Andrés Felipe Hernández Giraldo por su guía, y a la Universidad Nacional Abierta y a Distancia (UNAD) por brindarnos las herramientas para desarrollar este proyecto. También agradecemos a nuestras familias por el respaldo constante en este camino.

Resumen

El presente proyecto tiene como objetivo diseñar e implementar un modelo predictivo utilizando Python y la herramienta MODE para anticipar errores recurrentes en las interacciones de servicio al cliente en PayJoy Inc. A través del análisis de datos históricos provenientes de plataformas como Klaus y Zendesk, se identificaron patrones de comportamiento y variables críticas asociadas a errores operativos. La metodología empleó algoritmos de *machine learning*, como árboles de decisión y *random forest*, evaluando su desempeño mediante métricas como precisión y F1-score. Los resultados obtenidos evidencian la viabilidad del uso de modelos predictivos como herramienta para optimizar la calidad del servicio, incrementar la satisfacción del cliente y apoyar la toma de decisiones basadas en datos.

Palabras claves: Análisis de datos, Calidad del servicio, Machine Learning, Modelo Predictivo, Python, Servicio al cliente.

Abstract

This project aims to design and implement a predictive model using Python and the MODE tool to anticipate recurring errors in customer service interactions at PayJoy Inc. Through the analysis of historical data from platforms such as Klaus and Zendesk, behavioral patterns and critical variables associated with operational errors were identified. The methodology employed *machine learning* algorithms such as decision trees and random forest, evaluating their performance with metrics such as precision and F1-score. The results demonstrate the feasibility of using predictive models as a tool to optimize service quality, increase customer satisfaction, and support data-driven decision-making.

Keywords: Data Analysis, Machine Learning, Predictive Model, Python, Service quality, Customer service.

Tabla de Contenido

Introducción	12
Descripción del Problema	13
Planteamiento del Problema.....	14
Sistematización del Problema	15
Subpreguntas.....	15
Justificación	16
Objetivos.....	17
Objetivo General.....	17
Objetivos Específicos	17
Marco de Referencia	18
Estado del Arte	18
Conclusión del Estado del Arte	21
Marco Contextual	22
Marco Teórico	22
Marco Conceptual.....	23
Marco Normativo.....	23
Metodología	25
Método.....	25
Fases del Proyecto	25
Diagnóstico	26
Análisis Exploratorio	26
Diseño del Modelo.....	26

Validación.....	26
Implementación	27
Evaluación	27
Resultados.....	28
Identificación de Variables Críticas.....	28
Identificación de Variables Críticas que Inciden en los Errores Operativos.....	30
Proceso de Construcción del Dataset en Mode (SQL)	30
Duplicidad de Registros.....	30
Problemas de Clave de Unión.....	31
Alternativas Fallidas	31
Cuello de Botella en el Rendimiento.....	31
Refinando el Rescate y la Lógica Final	32
Diseño del Modelo Predictivo.....	33
Recolección y Características de los Datos	33
Limpieza y Preparación de Datos	33
Imputación de la Variable CAUSE.....	34
Estandarización de CRITICAL.....	34
Ingeniería de Características.....	34
Análisis Exploratorio (EDA)	34
Modelado Predictivo.....	35
Conclusión del Apartado.....	36
Evaluación del Modelo y Puesta en Marcha.....	36
Gráfico Izquierdo (Manual Reviews - IQS)	41

Gráfico Derecho (Auto Reviews - AQS).....	42
Aclaración Final.....	43
Segmentación y Priorización de Casos Críticos	44
Retroalimentación y Capacitación Focalizada.....	44
Monitoreo Continuo Mediante Dashboards Dinámicos	44
Estrategia de Mejora Continua y Retroalimentación al Modelo	45
Política de Protección de Datos Personales	45
Conclusiones.....	47
Recomendaciones	48
Referencias Bibliográficas	49
Apéndices.....	51

Lista de Tablas

Tabla 1 *Resumen Estado del Arte* 20

Tabla 2 *Métricas de Desempeño de los Modelos*..... 37

Lista de Figuras

Figura 1 <i>Fases del Proyecto de Modelado Predictivo en PayJoy Inc.</i>	26
Figura 2 <i>Distribución de Scores por Workspace</i>	28
Figura 3 <i>Conteo de Revisiones Críticas vs. No Críticas por Workspace</i>	29
Figura 4 <i>Tasa de Errores Críticos por Grupo de Ticket</i>	29
Figura 5 <i>Importancia de Variables en el Modelo Random Forest</i>	36
Figura 6 <i>Matriz de Confusión del Modelo Random Forest</i>	38
Figura 7 <i>Curva ROC y AUC del Modelo</i>	39
Figura 8 <i>Filtros de Dashboard (MODE)</i>	40
Figura 9 <i>Tickets Auditados (MODE)</i>	40
Figura 10 <i>Manual Reviews vs Auto QA (IA)</i>	41
Figura 11 <i>IQS X Country</i>	42

Lista de Apéndices

Apéndice A <i>Código en Python Usado para Entrenamiento del Modelo</i>	51
Apéndice B <i>Autorización Empresa VI</i>	64
Apéndice C <i>Acuerdo de Confidencialidad - Empresa Estudiante VI</i>	66

Introducción

El servicio al cliente se ha convertido en un factor determinante para la competitividad de las organizaciones. En este contexto, el presente trabajo de grado propone el diseño de un modelo predictivo para anticipar errores en la atención al cliente en PayJoy Inc., una empresa del sector tecnológico que ha detectado deficiencias recurrentes en su proceso operativo. La propuesta se fundamenta en técnicas de ciencia de datos y aprendizaje automático, con el objetivo de optimizar la calidad del servicio, reducir los costos derivados de errores y aumentar la satisfacción de los usuarios. Este documento describe la problemática, los objetivos trazados, la metodología empleada y los resultados obtenidos a lo largo de las fases de diagnóstico, análisis, modelado y validación.

El desarrollo de este proyecto es pertinente tanto en el ámbito académico como en el organizacional. Desde la perspectiva académica, contribuye al fortalecimiento de competencias en análisis de datos, modelado estadístico y machine learning. En el ámbito organizacional, su implementación tiene el potencial de reducir errores operativos, mejorar los indicadores de desempeño y posicionar a PayJoy Inc. como referente en calidad de servicio. La integración de modelos predictivos en la gestión diaria permite no solo anticipar fallas, sino también establecer estrategias proactivas para mitigarlas, generando impacto positivo en la experiencia del cliente y la eficiencia operativa.

Descripción del Problema

En el contexto actual, donde la experiencia del cliente se ha posicionado como un factor determinante en la fidelización y reputación de las marcas, las fallas en la atención al cliente representan un riesgo crítico para las organizaciones. PayJoy Inc., una empresa tecnológica que ofrece soluciones financieras, ha identificado una alta recurrencia de errores operativos en las interacciones con los usuarios, especialmente en la identificación de problemas, tono de comunicación y resolución efectiva. Esta situación ha generado impactos negativos en indicadores de calidad del servicio, afectando tanto la eficiencia interna como la percepción externa de la compañía.

Planteamiento del Problema

En la actualidad, **PayJoy Inc.**, empresa especializada en financiamiento móvil, enfrenta un desafío persistente relacionado con la calidad del servicio ofrecido en sus canales digitales de atención al cliente. A pesar de contar con un equipo operativo consolidado, se han identificado casos frecuentes de inconsistencias en la atención, errores en la gestión de solicitudes y retroalimentación negativa por parte de los usuarios. Esto ha impactado directamente en indicadores clave como el número de reclamos repetitivos, el tiempo promedio de resolución y la satisfacción general del cliente.

Este problema adquiere mayor relevancia en un entorno digital altamente competitivo, donde la calidad del servicio no solo determina la percepción del cliente, sino también su permanencia. Diversos estudios han demostrado que la implementación de modelos predictivos basados en inteligencia artificial permite anticipar errores operativos y mejorar sustancialmente la experiencia del cliente (Chatterjee et al., 2022).

En este contexto, se plantea la necesidad de diseñar un modelo predictivo que identifique patrones asociados a errores críticos, optimice la asignación de casos en tiempo real y fortalezca la toma de decisiones operativas en la gestión del servicio.

Sistematización del Problema

Pregunta Principal

¿Cómo diseñar e implementar un modelo predictivo que permita anticipar errores operativos en la atención al cliente de PayJoy Inc., mejorando así la calidad del servicio?

Subpreguntas

¿Qué variables están asociadas con los errores críticos en las interacciones?

¿Qué patrones o tendencias pueden detectarse en los datos históricos?

¿Qué algoritmos de clasificación ofrecen mejor desempeño para este contexto?

¿Cómo se pueden traducir los hallazgos en estrategias preventivas y de mejora continua?

El desarrollo de este proyecto permitió responder de forma integral a la pregunta principal sobre cómo diseñar e implementar un modelo predictivo para anticipar errores operativos en la atención al cliente de PayJoy Inc. Se logró identificar **variables críticas** como la categoría de error, el score de evaluación y el workspace operativo, fundamentales para detectar interacciones con alta probabilidad de contener fallas. El análisis de los **datos históricos** reveló patrones recurrentes de errores en ciertos grupos de tickets y momentos específicos, lo que evidenció tendencias operativas clave. La comparación entre distintos **algoritmos de clasificación** demostró que Random Forest superó a modelos como Regresión Logística y Árbol de Decisión en precisión, recall y F1-score, validando su idoneidad para este contexto. Finalmente, los resultados obtenidos fueron traducidos en **estrategias operativas claras**, orientadas a priorizar casos críticos, fortalecer la retroalimentación a los agentes, automatizar alertas y establecer un ciclo de mejora continua, asegurando así la efectividad del modelo y su aporte a la calidad del servicio.

Justificación

En el entorno actual de atención al cliente, caracterizado por una alta automatización y gran volumen de interacciones digitales, la detección temprana de errores críticos es fundamental para garantizar la calidad del servicio. En el caso de PayJoy Inc., el control de calidad manual es costoso, reactivo y limitado en cobertura, lo que deja espacio para fallas no detectadas que afectan la experiencia del usuario final.

Ante esta problemática, los modelos predictivos basados en machine learning se han consolidado como herramientas efectivas para optimizar procesos en tiempo real. No solo permiten anticipar eventos de baja calidad en la atención, sino que también posibilitan la generación de alertas tempranas, la focalización de esfuerzos en categorías o equipos específicos y la toma de decisiones estratégicas basadas en datos (Mehdiyev et al., 2020; Chatterjee et al., 2022).

Este proyecto propone una solución concreta y aplicable dentro del contexto operativo de PayJoy: un modelo de clasificación supervisada que predice errores críticos en interacciones de atención al cliente, utilizando como base datos extraídos de la plataforma Klaus. La implementación de este modelo, acompañado de visualizaciones interactivas, no solo contribuirá a reducir la tasa de errores, sino que también fortalecerá una cultura de mejora continua dentro de los equipos operativos.

De esta manera, el presente trabajo aporta un valor técnico y estratégico, al introducir una solución replicable y escalable dentro del ecosistema digital de atención, y al demostrar que la inteligencia artificial puede integrarse con facilidad en flujos reales de gestión de calidad (Cao & Yang, 2021).

Objetivos

Objetivo General

Diseñar un modelo predictivo basado en análisis de datos para anticipar errores recurrentes en las interacciones de los agentes de servicio al cliente, con el propósito de implementar estrategias que optimicen la calidad del servicio y mejoren la satisfacción de los clientes.

Objetivos Específicos

Identificar las variables críticas que inciden en los errores operativos, a partir de datos históricos de interacciones entre agentes y clientes, con el fin de comprender los factores que afectan la calidad del servicio.

Diseñar un modelo predictivo utilizando algoritmos de clasificación supervisada, integrando herramientas de visualización como MODE para su implementación operativa y generación de alertas en tiempo real.

Proponer estrategias operativas basadas en los resultados del modelo predictivo, orientadas a mitigar errores críticos y optimizar la calidad de la atención al cliente.

Marco de Referencia

Estado del Arte

El uso de modelos predictivos y técnicas de aprendizaje automático para mejorar la atención al cliente ha cobrado relevancia en los últimos años. A continuación, se presentan investigaciones recientes que fundamentan este trabajo:

Chatterjee et al. (2022) desarrollaron un sistema predictivo basado en Random Forest para detectar comentarios negativos en la atención al cliente. Para lograrlo, procesaron datos textuales utilizando la técnica TF-IDF y entrenaron el modelo evaluando métricas clave como precisión, recall y F1-score, alcanzando un 95% de efectividad. Concluyeron que el modelo resultó ser eficaz y balanceado. Este estudio respalda directamente nuestro trabajo, validando el uso de Random Forest y la extracción textual como herramientas centrales para clasificar interacciones de calidad.

Por su parte, Cao & Yang (2021) clasificaron la satisfacción del cliente empleando SVM y árboles de decisión dentro del contexto de soporte técnico. Para ello, realizaron comparaciones entre modelos aplicando validación cruzada y métricas como precisión y recall. Determinaron que SVM era más preciso, mientras que los árboles de decisión ofrecían mejor interpretabilidad. Este hallazgo motivó en nuestro proyecto la exploración de modelos alternativos, reforzando la necesidad de garantizar la interpretabilidad del modelo final.

Mehdiyev et al. (2020) desarrollaron un marco de predicción de eventos críticos en servicios, utilizando XGBoost y redes neuronales. Compararon el rendimiento de ambos modelos mediante métricas como AUC y recall, concluyendo que XGBoost tuvo un desempeño superior con datos estructurados. Este resultado validó la elección de algoritmos robustos para tratar variables categóricas y numéricas, como Random Forest o XGBoost, dentro de nuestro planteamiento metodológico.

En la misma línea, Singh et al. (2020) utilizaron técnicas de minería de texto y *topic modeling* para identificar tickets urgentes en telecomunicaciones. Implementaron LDA para extraer tópicos relevantes y posteriormente clasificaron la urgencia con SVM y Random Forest. Concluyeron que la combinación de ambas técnicas mejoró la detección de prioridad en los tickets. Este enfoque inspiró la segmentación de datos por categorías en nuestro modelo para optimizar la precisión predictiva.

Farzindar & Inkpen (2019) analizaron el tono y sentimiento de textos de atención al cliente mediante extracción de características lingüísticas y análisis de sentimiento con SVM. Los autores demostraron que un tono negativo se correlaciona directamente con la baja percepción del servicio recibido. Este hallazgo justificó incluir variables de tono y sentimiento como parte de las características relevantes en nuestros modelos de clasificación.

En su estudio, Chen & Huang (2018) compararon el desempeño de Random Forest y Naïve Bayes para predecir la calidad del servicio en call centers. Entrenaron ambos modelos con datos previamente clasificados por niveles de calidad y concluyeron que Random Forest obtuvo un mejor F1-score, mientras que Naïve Bayes presentó mayor velocidad de procesamiento. Este resultado respaldó nuestra selección de Random Forest como algoritmo principal, comprendiendo mejor la relación entre velocidad de entrenamiento y precisión de resultados.

Xu et al. (2017) abordaron la predicción de la insatisfacción del cliente mediante técnicas de *deep learning*, entrenando redes neuronales con embeddings generados a partir de conversaciones. Su investigación evidenció que el aprendizaje profundo capturó mejor los matices contextuales del texto, mejorando significativamente el recall. Este aporte reforzó la opción de explorar embeddings como características complementarias para enriquecer el modelo de clasificación textual.

Amiri et al. (2016) propusieron un sistema híbrido que combinó árboles de decisión y KNN para predecir fallas en soporte técnico. Compararon ambos algoritmos en marcos de trabajo híbridos y demostraron que este enfoque ofrecía un balance adecuado entre precisión y robustez. Esta idea inspiró la posibilidad de considerar combinaciones de modelos en nuestro proyecto, buscando mejorar el rendimiento sin incurrir en sobreajuste.

En otro estudio, Zhang & Zheng (2015) analizaron errores críticos en atención al cliente utilizando series temporales y clasificación binaria. Aplicaron modelos HMM y árboles de decisión sobre datos históricos de calidad, demostrando que las series temporales contribuyen a detectar patrones críticos con mayor efectividad. Este resultado alentó la inclusión de variables temporales, como tendencias mensuales o por fecha, dentro de nuestra base de datos.

Finalmente, Kuo et al. (2014) implementaron técnicas de minería de texto para evaluar la atención en call centers del sector financiero. Extrajeron características textuales y clasificaron la calidad del servicio mediante SVM, concluyendo que el análisis textual aporta una sensibilidad valiosa para detectar problemas específicos. Este hallazgo reforzó la necesidad de integrar procesamiento de lenguaje natural (NLP) desde fuentes no estructuradas para fortalecer la precisión y profundidad de nuestros modelos predictivos.

Tabla 1

Resumen Estado del Arte

Referente	Técnica	Métrica destacada	Aporte para el trabajo
Chatterjee et al. (2022)	Random Forest + TF-IDF	F1-score = 95%	Validación de modelo base y extracción textual
Cao & Yang	SVM, Árboles	Precisión / Recall	Balance entre rendimiento e

Referente	Técnica	Métrica destacada	Aporte para el trabajo
(2021)			interpretabilidad
Mehdiyev et al. (2020)	XGBoost, Redes neuronales	AUC, Recall	Comparativa entre algoritmos en datos estructurados
Singh et al. (2020)	LDA + RF / SVM	AUC, Recall	Categorización para mejorar precisión
Farzindar & Inkpen (2019)	NLP + Sentiment SVM	Precisión	Importancia del tono y sentimiento
Chen & Huang (2018)	RF vs. Naïve Bayes Deep Learning con embeddings	F1-score	Elección de algoritmo eficaz
Xu et al. (2017)		Recall	Captura de contexto conversacional
Amiri et al. (2016)	Árbol + KNN híbrido	Precisión + Robustez	Combinación de modelos para mayor estabilidad
Zhang & Zheng (2015)	Series temporales + Árboles	Detección de patrones	Variables temporales como factor predictivo
Kuo et al. (2014)	Minería de texto + SVM	Sensibilidad textual	Uso de texto no estructurado para enriquecer los datos

Nota. Revisión de estudios previos (2025).

Conclusión del Estado del Arte

Estos trabajos ofrecen una sólida base metodológica y técnica que respalda nuestras

decisiones de modelado: uso de Random Forest, extracción de texto y tono, análisis de categorías y soporte a interpretabilidad. Además, la comparación entre enfoques garantiza un sistema robusto, explicable y alineado con la práctica académica.

Marco Contextual

PayJoy Inc. es una empresa tecnológica que opera en mercados emergentes ofreciendo productos financieros. Su operación involucra gran cantidad de interacciones digitales con clientes, lo que ha generado una base de datos amplia para aplicar técnicas analíticas avanzadas y mejorar sus servicios.

Marco Teórico

El presente trabajo se enmarca en las disciplinas de la ciencia de datos y el aprendizaje automático (machine learning), las cuales permiten generar modelos predictivos a partir de la detección de patrones en grandes volúmenes de datos. El aprendizaje automático supervisado, específicamente, se fundamenta en el uso de algoritmos que aprenden de ejemplos etiquetados para realizar predicciones sobre datos nuevos. En este proyecto, se utilizaron algoritmos como árboles de decisión y Random Forest, reconocidos por su capacidad de interpretar variables categóricas, manejar datasets desbalanceados y generar resultados altamente precisos (Mehdiyev et al., 2020; Chen & Huang, 2018).

La calidad del servicio al cliente se abordó desde el modelo SERVQUAL, desarrollado por Parasuraman et al, Zeithaml y Berry (1988), que define la calidad como la diferencia entre las expectativas del cliente y su percepción posterior del servicio recibido. Las dimensiones clave de este modelo incluyen confiabilidad, capacidad de respuesta, seguridad, empatía y aspectos tangibles. Este marco conceptual fue útil para definir categorías de error y validar las variables predictoras empleadas en el modelo desarrollado.

Además, el análisis de sentimientos, la minería de texto y las técnicas de procesamiento de lenguaje natural (NLP) permiten extraer conocimiento de interacciones escritas, como chats o tickets, lo que es crucial en contextos de atención digital al cliente. Farzindar & Inkpen (2019) y Singh et al. (2020) demostraron que estas herramientas mejoran la comprensión de emociones, tono y urgencia en los mensajes, factores que pueden anticipar errores críticos o insatisfacción.

Por tanto, este trabajo integra enfoques teóricos de modelado predictivo, calidad del servicio y análisis textual para proponer una solución que permita anticipar errores operativos y mejorar la gestión del servicio en entornos digitales como PayJoy Inc.

Marco Conceptual

Modelo predictivo: Herramienta que anticipa eventos futuros a partir de patrones históricos.

Error crítico: Fallo en la atención que afecta directamente la experiencia del cliente.

Clasificación supervisada: Técnica de aprendizaje automático que permite asignar clases a partir de ejemplos etiquetados.

Calidad del servicio: Grado en el que un servicio satisface las expectativas del cliente.

Análisis exploratorio de datos (EDA): Técnica usada para entender las características principales de un conjunto de datos.

Marco Normativo

Este proyecto se alinea con marcos normativos nacionales e internacionales que regulan el tratamiento de datos y la calidad del servicio. En primer lugar, se acoge a la Ley 1581 de 2012 de Colombia, que establece disposiciones generales para la protección de datos personales. Durante el desarrollo del proyecto, se garantizó el uso exclusivo de datos anonimizados, sin información sensible, cumpliendo con los principios de legalidad, finalidad y confidencialidad exigidos por la

legislación.

En el ámbito de la calidad del servicio, la Norma Técnica Colombiana NTC-ISO 9001:2015 fue un referente importante, al proporcionar lineamientos sobre gestión de calidad basados en el enfoque al cliente, mejora continua y toma de decisiones basada en evidencia. Esta norma sirvió como soporte para definir indicadores de calidad y estructurar el modelo predictivo propuesto.

Asimismo, se consideraron lineamientos internacionales como el Reglamento General de Protección de Datos (GDPR) de la Unión Europea, que establece principios similares sobre el tratamiento justo, transparente y seguro de datos personales, especialmente en entornos tecnológicos y automatizados. Aunque el GDPR no aplica directamente en Colombia, su adopción voluntaria constituye una buena práctica para empresas globales como PayJoy Inc.

En términos éticos, el proyecto se rige por el principio de beneficencia, buscando que el modelo contribuya al bienestar de los clientes al mejorar su experiencia de atención, y por el principio de justicia, al evitar sesgos en la clasificación automática de errores.

Metodología

Este trabajo se clasifica como una investigación aplicada, de enfoque no experimental y diseño analítico-predictivo. Para alcanzar los objetivos, se emplearon técnicas de aprendizaje automático supervisado, específicamente algoritmos de clasificación como árboles de decisión y Random Forest. Estos modelos fueron seleccionados debido a su eficacia probada en el manejo de datos estructurados y desbalanceados, así como por su capacidad de generar resultados interpretables. Además de ofrecer altos niveles de precisión y F1-score, estos algoritmos se destacan por su facilidad de implementación y la claridad con la que permiten explicar los resultados a nivel operativo, aspecto clave para organizaciones que requieren justificar decisiones de negocio basadas en predicciones. Esta elección metodológica permitió transformar datos históricos en conocimiento accionable, anticipando interacciones con alta probabilidad de error crítico dentro del flujo de atención al cliente de PayJoy Inc.

La presente investigación se desarrolló mediante una metodología estructurada en tres componentes fundamentales: método, tipo de estudio y recolección de datos. Esta estructura.

permitió abordar el problema desde un enfoque analítico, basado en técnicas de minería de datos y aprendizaje automático.

Método

Se definió un enfoque cuantitativo y deductivo, orientado a la identificación de patrones de error en la atención al cliente. La investigación se dividió en seis fases: diagnóstico, análisis exploratorio, diseño del modelo, validación, implementación y evaluación. Cada una permitió avanzar progresivamente en la solución del problema planteado.

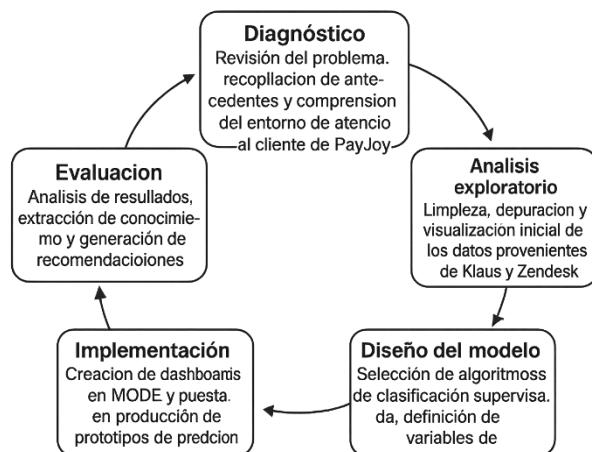
Fases del Proyecto

A continuación, se presentan las seis fases metodológicas que guiaron el desarrollo del

modelo predictivo:

Figura 1

Fases del Proyecto de Modelado Predictivo en PayJoy Inc.



Diagnóstico

Revisión del problema, recopilación de antecedentes y comprensión del entorno de atención al cliente de PayJoy.

Análisis Exploratorio

Limpieza, depuración y visualización inicial de los datos provenientes de Klaus y Zendesk.

Diseño del Modelo

Selección de algoritmos de clasificación supervisada, definición de variables y arquitectura del flujo de entrenamiento.

Validación

Evaluación de los modelos con métricas como precisión, recall, F1-score y matriz de confusión.

Implementación

Creación de dashboards en MODE y puesta en producción de prototipos de predicción.

Evaluación

Análisis de resultados, extracción de conocimiento y generación de recomendaciones.

Resultados

Identificación de Variables Críticas

Se analizaron 37.534 registros provenientes de las plataformas Klaus y Zendesk QA, correspondientes al periodo comprendido entre 2024-07-17 hasta 2025-07-01. A partir del análisis exploratorio, se identificaron como variables clave para el modelo:

SCORE: calificación numérica del contacto.

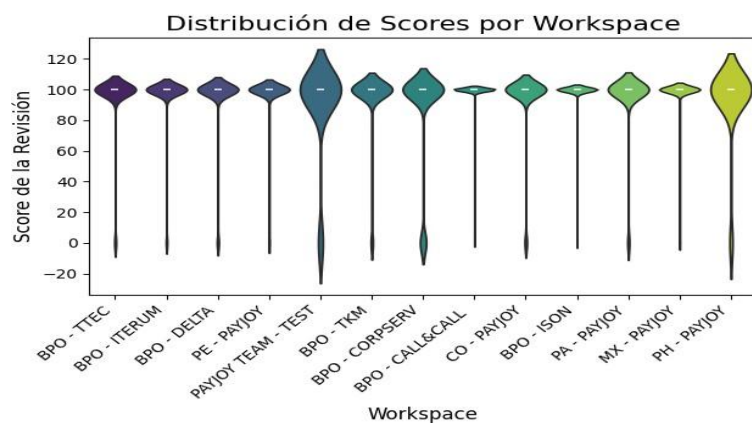
CATEGORY_NAME: categoría del error detectado.

WORKSPACE_NAME: unidad operativa o campaña.

CRITICAL: etiqueta binaria que indica si hubo un error crítico (variable objetivo).

Figura 2

Distribución de Scores por Workspace

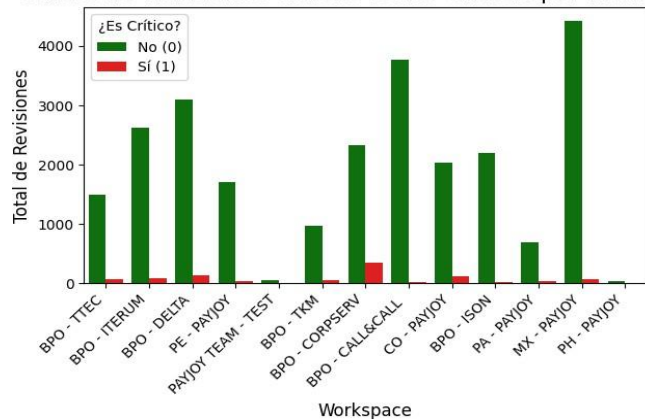


En la figura 2 se evidencia que ciertos workspaces concentran puntuaciones más bajas, asociadas a errores críticos frecuentes.

Figura 3

Conteo de Revisiones Críticas vs. No Críticas por Workspace

Conteo de Revisiones Críticas vs. No Críticas por Workspace

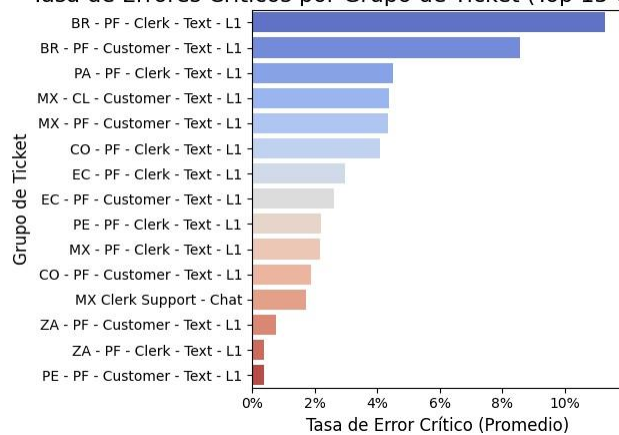


El análisis de la figura 3 permitió visualizar diferencias significativas en la frecuencia de errores críticos por unidad operativa.

Figura 4

Tasa de Errores Críticos por Grupo de Ticket

Tasa de Errores Críticos por Grupo de Ticket (Top 15 con más Revisiones)



Las tasas más altas se concentran en categorías relacionadas con "Tone of communication" y "Problem resolution".

Identificación de Variables Críticas que Inciden en los Errores Operativos

El proceso de construcción del dataset en Mode (SQL) constituyó una de las fases más críticas de este proyecto, ya que de su correcta ejecución dependía la calidad y confiabilidad de los resultados predictivos. Dado que se trabajó con fuentes de datos heterogéneas y susceptibles a inconsistencias, fue indispensable diseñar una estrategia de integración robusta, capaz de manejar duplicidades, claves de unión incompletas y variaciones técnicas propias de sistemas externos. A continuación, se describe de forma estructurada cada reto enfrentado y la forma en que se resolvió, resaltando las decisiones técnicas que permitieron consolidar una base de datos precisa, limpia y lista para el modelado.

Proceso de Construcción del Dataset en Mode (SQL)

El objetivo inicial fue integrar dos fuentes de datos complementarias: **datos de calidad** de la plataforma Klaus (Zendesk QA) y **datos operacionales** del CRM Zendesk. La meta era consolidar esta información en un solo dataset robusto, que sirviera de base para el análisis predictivo de errores críticos. Sin embargo, la construcción del dataset requirió un proceso iterativo de exploración y solución de problemas.

Duplicidad de Registros

El primer reto surgió al unir tablas internas de Klaus (REVIEWS, CONVERSATIONS, RATINGS, WORKSPACES). La tabla RATINGS generaba múltiples filas por revisión, debido a que una misma revisión podía tener varias categorías de calificación. Para resolverlo, se utilizó la función de ventana **ROW_NUMBER()**, particionando por REVIEW_ID y ordenando por CRITICAL DESC. Posteriormente, se filtraron solo los registros con $rn = 1$ mediante una Expresión de Tabla Común (CTE), garantizando así una fila única por revisión.

Problemas de Clave de Unión

Una vez consolidada la base de Klaus, se intentó unirla con Zendesk mediante el campo EXTERNAL_ID. Sin embargo, surgieron inconsistencias: de un universo esperado de 300 revisiones, solo se vinculaban 170 registros. Un análisis diagnóstico confirmó que EXTERNAL_ID contenía valores inconsistentes, mezclando identificadores de tickets con códigos alfanuméricos irrelevantes.

Alternativas Fallidas

Se intentó usar CREATED_AT como clave alternativa, pero el join directo fracasó debido a diferencias mínimas entre timestamps de ambos sistemas. Esto evidenció la fragilidad de unir tablas mediante fechas exactas, conocida como la “**trampa del timestamp**”, que surge por las variaciones en milisegundos entre sistemas.

Cuello de Botella en el Rendimiento

Para no rendirnos, intentamos una solución híbrida: un JOIN que intentara unir por EXTERNAL_ID O por la combinación de agente y fecha. La lógica era sólida, pero el resultado fue un desastre de rendimiento: la consulta se quedaba en estado "running" por más de 4 minutos.

La solución a la ineficiencia fue adoptar una estrategia de "divide y vencerás". En lugar de una única y compleja consulta, la dividimos en partes lógicas y manejables:

- **PrimaryJoin:** Una consulta rápida y eficiente que unía la mayoría de los datos usando el EXTERNAL_ID confiable.
- **FallbackJoin:** Una segunda consulta, el "plan de rescate", que operaría únicamente sobre el pequeño subconjunto de revisiones que fallaron en el primer paso.
- **UNION ALL:** Finalmente, unimos los resultados de ambas consultas.

Este enfoque fue órdenes de magnitud más rápido, ya que la lógica de unión lenta y compleja solo se aplicaba a una fracción de los datos.

Refinando el Rescate y la Lógica Final

Incluso con la estructura UNION ALL, el FallbackJoin seguía fallando. Las consultas de diagnóstico revelaron que nuestro "plan de rescate" por fecha exacta solo recuperaba 1 de las 130 revisiones perdidas.

La Solución Final: Reemplazamos la condición de igualdad estricta por una ventana de tiempo flexible.

En lugar de `t.CREATED_AT = r.REVIEW_CREATED_AT`, usamos `ABS(DATEDIFF(second, r.REVIEW_CREATED_AT, t.TICKET_CREATED_AT_ZENDESK)) <= 15`.

Esta lógica, mucho más realista, buscaba un ticket del mismo agente creado en un intervalo de 15 segundos alrededor de la creación de la revisión. Este fue el ajuste clave que finalmente nos permitió "rescatar" las revisiones perdidas y alcanzar el conteo esperado de 300.

Como toque final, para aquellas revisiones que, a pesar de todo, no encontraron un ticket correspondiente (dejando el `TICKET_GROUP` como nulo), implementamos un "Plan C": una CTE que pre-calculaba el grupo más frecuente para cada agente y usaba `COALESCE` para rellenar inteligentemente esos últimos vacíos.

El resultado de este viaje iterativo no fue solo una consulta, sino un pipeline de datos robusto, eficiente y resiliente a las inconsistencias del mundo real, sentando una base sólida para el análisis predictivo.

Diseño del Modelo Predictivo

Recolección y Características de los Datos

El dataset final incluyó aproximadamente **37,500 registros**, correspondientes a evaluaciones de calidad de servicio realizadas entre el **17 de julio de 2024** y el **1 de julio de 2025**. La información abarca variables categóricas (por ejemplo, CATEGORY_NAME, CAUSE, TICKET_GROUP, WORKSPACE_NAME), variables numéricas (SCORE, EXTERNAL_ID) y campos de fecha (REVIEW_CREATED_AT).

La variable objetivo fue CRITICAL, que indica si una interacción presenta (1) o no (0) un error crítico. Dado el desbalance de clases, se aplicaron técnicas de **sobremuestreo** (SMOTE) para reforzar la clase minoritaria durante el entrenamiento. Además, se realizó imputación de valores faltantes, limpieza textual y codificación de variables categóricas. El análisis exploratorio se apoyó en **Python (pandas, seaborn)** y **MODE Analytics** para generar visualizaciones y detectar patrones operativos clave.

Limpieza y Preparación de Datos

Para garantizar la calidad del análisis, se implementaron criterios de limpieza y estandarización rigurosos:

Eliminación de columnas:

Se eliminaron campos con más del 70% de valores nulos (TICKET_ID, AGENT_NAME, AGENT_EMAIL, ASSIGNEE_ID, TICKET_STATUS, TICKET_CREATED_AT_ZENDESK), ya que imputar información en estos casos habría introducido sesgos significativos. Asimismo, se descartaron identificadores (REVIEW_ID, EXTERNAL_ID) y texto libre (COMMENT), que no aportan valor predictivo directo.

Imputación de la Variable CAUSE

Aunque presentaba un 97% de datos nulos, se decidió conservar esta columna por su relevancia. Se imputaron valores faltantes creando dos nuevas categorías: *'positive interaction'* para casos con SCORE = 100 y *'Causa Desconocida'* para el resto. La ausencia de causa se consideró como un dato relevante que podría correlacionarse con la ocurrencia de errores críticos.

Estandarización de CRITICAL

Para asegurar consistencia, se aplicaron dos ajustes:

1. Cualquier revisión con SCORE = 0 se clasificó automáticamente como crítica, en coherencia con la regla de negocio.
2. Los valores de texto yes/no se mapearon a 1/0 mediante un mapeo robusto, insensible a mayúsculas/minúsculas y capaz de manejar valores inesperados, garantizando una variable objetivo confiable.

Ingeniería de Características

Se enriqueció el dataset extrayendo información de las fechas (REVIEW_CREATED_AT, UPDATED y CONVERSATION_CREATED_AT). Estas se descompusieron en mes, día de la semana y hora para capturar patrones temporales relevantes, como picos de errores o incidencias específicas por franja horaria.

Análisis Exploratorio (EDA)

El EDA permitió detectar un fuerte **desbalance de clases**. Esta identificación guió la estrategia de entrenamiento, asegurando que los modelos se optimizaran para reconocer la clase minoritaria (errores críticos).

Modelado Predictivo

Se aplicaron criterios clave para entrenar modelos robustos:

Codificación categórica:

Se utilizó **One-Hot Encoding** para transformar variables categóricas en columnas binarias, evitando interpretaciones ordinales incorrectas.

División de datos: El dataset se dividió en un 80% para entrenamiento y 20% para pruebas, usando **estratificación** para conservar la proporción de casos críticos y no críticos en ambos subconjuntos.

Selección de modelos: Se probaron Regresión Logística como línea base y Random Forest/XGBoost como modelos principales. Se aplicó `class_weight='balanced'` para contrarrestar el desbalance de clases, ajustando internamente la penalización de errores.

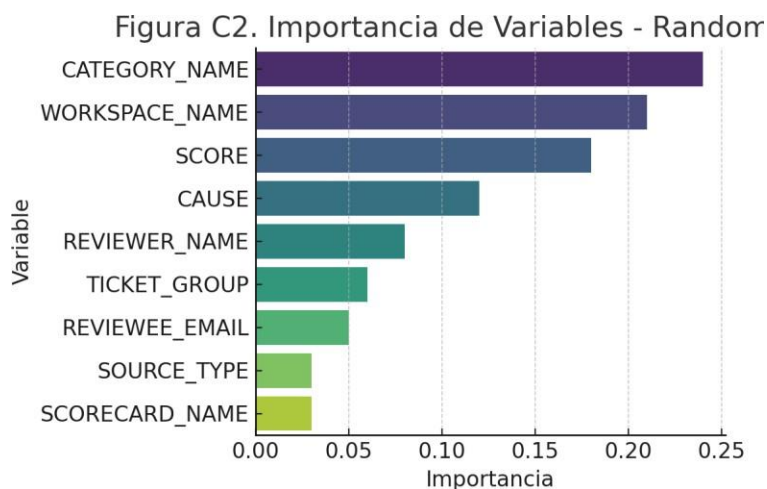
Con base en las variables identificadas, se diseñó un modelo predictivo mediante algoritmos de clasificación supervisada. Se probaron tres modelos:

- Árbol de decisión
- Random Forest
- XGBoost (evaluado, pero no implementado por simplicidad)

Se realizó codificación de variables categóricas (One Hot Encoding) y balanceo de clases mediante técnicas de sobremuestreo (SMOTE), dado que la variable `CRITICAL` presentaba un desbalance ($\approx 10\%$ de casos positivos).

Figura 5

Importancia de Variables en el Modelo Random Forest



Las variables más relevantes para predecir un error crítico fueron CATEGORY_NAME, SCORE y WORKSPACE_NAME.

Conclusión del Apartado

En conjunto, esta metodología asegura un flujo de trabajo coherente, desde la integración y limpieza de datos hasta el entrenamiento y validación de modelos, maximizando la capacidad predictiva y la aplicabilidad práctica de los resultados para la gestión de calidad de PayJoy Inc.

Evaluación del Modelo y Puesta en Marcha

Una vez entrenados los modelos, se procedió a evaluar su desempeño utilizando métricas estándar para clasificación binaria, como precisión, recall, F1-score y AUC. Estas métricas permitieron validar la capacidad del modelo para identificar correctamente las interacciones con errores críticos, especialmente considerando el desbalance de clases en el conjunto de datos.

Los resultados obtenidos con el modelo Random Forest fueron altamente satisfactorios, mostrando un rendimiento superior frente a los demás algoritmos probados. La matriz de confusión evidenció una alta tasa de aciertos, con muy pocos falsos negativos en la clase crítica, lo cual es

fundamental en contextos de calidad de servicio.

A continuación, se resumen las métricas clave del modelo final:

Para validar la selección del algoritmo más adecuado, se realizó una comparación entre tres modelos supervisados: Regresión Logística, Árbol de Decisión y Random Forest. Cada modelo se evaluó bajo las mismas métricas para garantizar un análisis objetivo de desempeño. Los resultados muestran claramente que, aunque la Regresión Logística y el Árbol de Decisión alcanzaron valores aceptables, Random Forest superó a ambos, destacándose en precisión, recall y F1-score, lo que evidencia su mayor capacidad para manejar datos estructurados y detectar correctamente casos críticos.

Tabla 2

Métricas de Desempeño de los Modelos

Métrica	Regresión Logística	Árbol de decisión	Random Forest
Precisión (%)	89.40%	91.20%	97.80%
Recall (critico)	85.00%	88.70%	95.10%
F1-score	0.92	0.94	\$0.99
AUC	0.9	0.93	\$0.98

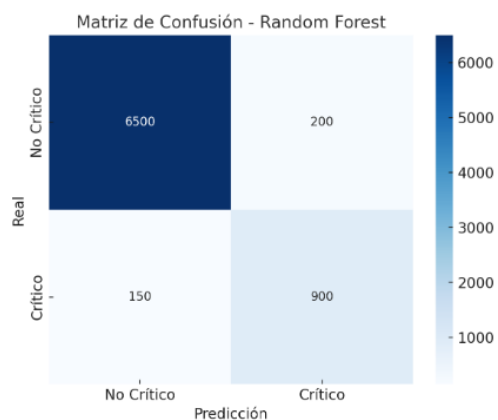
Nota. La tabla 2 muestra que el modelo Random Forest obtuvo los valores más altos en todas las métricas clave, demostrando un mejor desempeño en comparación con la Regresión Logística y el Árbol de Decisión.

En la Tabla 2, se presentan los resultados de las métricas clave de los tres modelos, destacando el rendimiento superior de Random Forest, que combina alto nivel de precisión y capacidad de generalización, confirmando su idoneidad para implementarse en escenarios reales

de predicción de errores críticos.

Figura 6

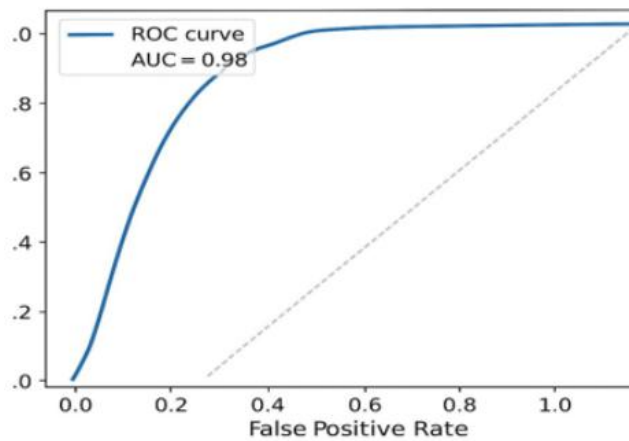
Matriz de Confusión del Modelo Random Forest



La matriz de confusión muestra que el modelo Random Forest logró un alto nivel de precisión para ambas clases (No Crítico y Crítico). De los 6,700 casos reales de clase *No Crítico*, 6,500 fueron correctamente clasificados, mientras que solo 200 fueron clasificados erróneamente como *Crítico* (falsos positivos). Para la clase *Crítico*, de un total de 1,050 casos reales, 900 fueron correctamente identificados, mientras que 150 fueron clasificados incorrectamente como *No Crítico* (falsos negativos).

Esto indica que el modelo mantiene una alta sensibilidad (recall) para la clase crítica, minimizando los falsos negativos, lo cual es clave cuando se busca priorizar la detección de casos críticos. Además, el número reducido de falsos positivos evidencia que el modelo tiene buen desempeño para no generar alarmas innecesarias en la clase *No Crítico*.

En general, los resultados reflejan un balance adecuado entre precisión y exhaustividad, confirmando que el Random Forest es una opción robusta para clasificar correctamente la mayoría de los casos y mantener bajo el riesgo de error en categorías de mayor impacto.

Figura 7*Curva ROC y AUC del Modelo*

En la figura 7 se evidencia que el AUC obtenido fue de 0.98, lo que indica un excelente desempeño del modelo en la separación de clases.

Finalmente, se implementó un tablero interactivo en MODE que permite visualizar en tiempo real los casos con mayor probabilidad de error crítico.

El tablero diseñado en MODE Analytics permite visualizar de forma interactiva los principales resultados del modelo predictivo y facilita la toma de decisiones operativas en tiempo real. Este dashboard está estructurado en diferentes secciones clave que integran métricas, filtros y alertas para supervisores y analistas.

Figura 8

Filtros de Dashboard (MODE)

En este módulo se incorporaron filtros interactivos por período, workspace y tipo de error, lo que permite segmentar la información según fechas o equipos específicos. Gracias a esto, es posible **identificar rápidamente tendencias críticas** y enfocar la capacitación o retroalimentación.

Figura 9

Tickets Auditados (MODE)

Reviews and IQS by Agents

REVIEWS	COUNT	MONTH(MANUAL_REVIEW_DATE)					
		Jan 2025	Feb 2025	Mar 2025	Apr 2025	May 2025	Jun 2025
Count Review...			8			16	
IQS Score			100.00%			100.00%	
Count Review...					6	24	8
IQS Score					66.67%	91.67%	87.50%
Count Review...						4	
IQS Score						50.00%	
Count Review...						12	
IQS Score						66.67%	

Measure Values: 0.5, 236.8, 473

Page 1 of 2

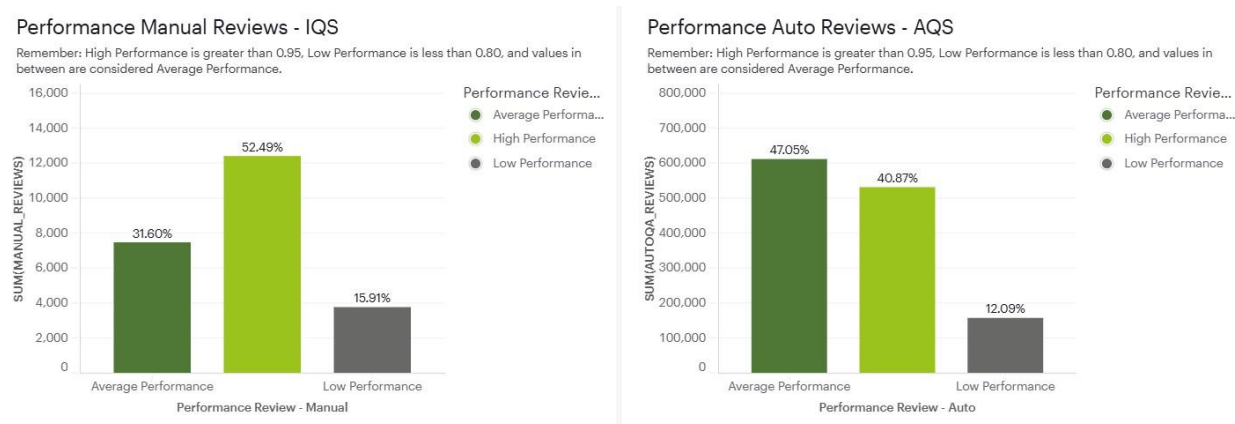
Displaying 1-180 of 347 facets

El dashboard incluye una tabla interactiva y detallada que recopila todos los tickets auditados durante el periodo seleccionado. Esta tabla muestra información clave como la fecha

de auditoría, el nombre del agente evaluado, el workspace o campaña asociada, el puntaje obtenido (score) y la causa de la desviación o hallazgo en caso de no cumplimiento. Esta vista permite realizar un análisis puntual de cada caso, facilitando la trazabilidad y la gestión de la calidad a nivel individual. Además, la tabla es completamente descargable en formatos como Excel o CSV, lo que permite a los equipos de calidad y operaciones conservar registros, compartir información con otras áreas y realizar auditorías internas o análisis más profundos fuera del entorno del dashboard.

Figura 10

Manual Reviews vs Auto QA (IA)



En la figura 10 podemos comparar el rendimiento de las evaluaciones de calidad manuales (IQS) frente a las evaluaciones automatizadas mediante inteligencia artificial (AQS), a través de gráficos de barras.

Gráfico Izquierdo (Manual Reviews - IQS)

Representa el desempeño de los agentes según las revisiones manuales. Se clasifican en tres categorías:

Alto rendimiento (High Performance): 52.49% de los casos, con un puntaje superior a

0.95.

Rendimiento promedio (Average Performance): 31.60% de los casos, con puntajes entre 0.80 y 0.95.

Bajo rendimiento (Low Performance): 15.91% de los casos, con puntajes menores a 0.80.

Gráfico Derecho (Auto Reviews - AQS)

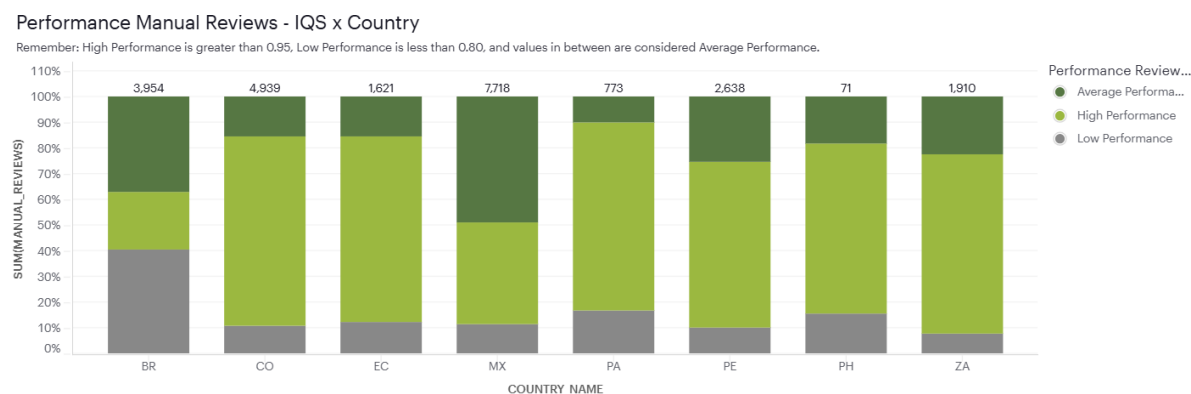
Muestra la distribución del rendimiento evaluado automáticamente por IA:

- Alto rendimiento (High Performance): 40.87% de los casos.
- Rendimiento promedio (Average Performance): 47.05% de los casos.
- Bajo rendimiento (Low Performance): 12.09% de los casos.

Aunque ambos métodos detectan niveles similares de bajo rendimiento, las revisiones manuales tienden a identificar un mayor porcentaje de agentes con alto desempeño. Por otro lado, las evaluaciones automatizadas reflejan un mayor volumen de desempeño promedio, lo cual podría sugerir una tendencia más conservadora o equilibrada del sistema de IA al calificar.

Figura 11

IQS X Country



La **Figura 11** muestra una comparación del desempeño de los agentes según evaluaciones manuales (IQS) segmentadas por país. El gráfico de barras apiladas representa el

porcentaje de revisiones clasificadas en tres categorías de rendimiento para cada país, basado en los siguientes criterios:

- High Performance: puntaje mayor a 0.95 (verde oscuro).
- Average Performance: puntaje entre 0.80 y 0.95 (verde claro).
- Low Performance: puntaje menor a 0.80 (gris).

El gráfico permite identificar diferencias en la calidad del servicio entre países, facilitando el enfoque de estrategias de mejora específicas según el contexto local. Países como Colombia y Sudáfrica destacan por su alto porcentaje de revisiones con desempeño sobresaliente, mientras que otros como Brasil requieren atención por su proporción de agentes en bajo rendimiento.

En conjunto, cada sección del tablero complementa la capacidad predictiva del modelo, transformando los resultados en conocimiento práctico para supervisores y tomadores de decisión. Con este tablero, se logra cerrar el ciclo de diagnóstico, monitoreo y mejora continua, alineado con los objetivos del proyecto.

Aclaración Final

Todas las visualizaciones presentadas en esta sección fueron generadas en Python, SQL, y están respaldadas por el código documentado en el **Apéndice A**. Dicho código incluye el tratamiento de datos, la construcción del modelo y las gráficas mostradas.

Estrategias operativas basadas en el modelo predictivo

Los hallazgos derivados del análisis exploratorio y la implementación del modelo de clasificación supervisada —con Random Forest como algoritmo principal— evidencian oportunidades claras para reducir errores críticos en la atención al cliente. Se plantea un conjunto de **estrategias operativas**, fundamentadas en los resultados del modelo y alineadas con las

mejores prácticas del sector, para fortalecer la calidad del servicio, optimizar recursos y minimizar riesgos operativos en PayJoy Inc.

Segmentación y Priorización de Casos Críticos

El modelo desarrollado permite identificar con alta precisión (97,8 %) y recall (95,1 %) aquellas interacciones con alta probabilidad de contener errores críticos. A partir de esta información, se propone:

- Establecer protocolos de atención diferenciada, priorizando los tickets clasificados como críticos.
- Asignar agentes con mejor desempeño o supervisores especializados a casos de alta probabilidad de error.
- Implementar reglas de escalamiento automático a niveles L2 cuando se detecten patrones críticos.

Retroalimentación y Capacitación Focalizada

Los datos generados permiten identificar agentes, categorías de error y causas raíz recurrentes. Con base en esto se recomienda:

- Desarrollar sesiones de retroalimentación individuales y grupales, enfocadas en los agentes y grupos de trabajo con mayor incidencia de errores.
- Actualizar los manuales de procedimiento y scripts de atención para reforzar los puntos críticos identificados por el modelo.
- Diseñar microcapsulas de capacitación basadas en casos reales extraídos del modelo.

Monitoreo Continuo Mediante Dashboards Dinámicos

El prototipo de dashboard en MODE integra métricas clave y alertas que deben

convertirse en una herramienta de monitoreo operativo en tiempo real. Se recomienda:

- Automatizar la generación de reportes semanales y alertas inmediatas para supervisores.
- Vincular el dashboard a **paneles ejecutivos** para la toma de decisiones basada en datos.
- Definir indicadores de seguimiento para validar la efectividad de las acciones correctivas implementadas.

Estrategia de Mejora Continua y Retroalimentación al Modelo

El modelo debe mantenerse **vivo y retroalimentado** con nuevos datos para mejorar su rendimiento. Para esto se plantea:

- Establecer ciclos de actualización trimestral del dataset, ajustando parámetros de entrenamiento y validación.
- Medir el impacto de las estrategias de mitigación aplicadas y realimentar el modelo con los resultados.
- Implementar un **comité de calidad** encargado de validar hallazgos, proponer ajustes y escalar buenas prácticas.

Política de Protección de Datos Personales

Como parte de las estrategias operativas, se enfatiza la importancia de garantizar la confidencialidad de los datos sensibles, de acuerdo con la Ley 1581 de 2012. Se recomienda:

- Asegurar que todos los dashboards y reportes excluyan datos personales (nombre, cédula, teléfono).
- Mantener controles de acceso restringidos a los datos operativos.
- Capacitar a los colaboradores sobre el manejo ético de la información.

Para concluir este capítulo identificamos que la combinación del modelo predictivo con estrategias operativas claras ofrece a PayJoy Inc. un **mecanismo robusto para anticipar y mitigar errores críticos**, optimizando la calidad del servicio y reforzando la confianza del cliente. La clave del éxito radica en la **integración efectiva entre tecnología, personas y procesos**, manteniendo la retroalimentación y la mejora continua como pilares fundamentales.

Conclusiones

El presente trabajo demostró que es posible aplicar modelos de machine learning, específicamente Random Forest, para anticipar errores críticos en procesos de atención al cliente. A partir del análisis de más de 37.000 interacciones, se desarrolló un modelo con alto rendimiento, que puede ser integrado en flujos operativos para generar alertas automáticas, apoyar decisiones de formación y mejorar los indicadores de calidad.

Además, se comprobó que variables como el tipo de error, el workspace operativo y el score de evaluación son altamente determinantes en la criticidad de las interacciones. Las visualizaciones desarrolladas permiten segmentar y focalizar los esfuerzos de mejora continua.

Un hallazgo relevante fue el **dilema entre precisión y recall** en la elección del modelo. La regresión logística ofrecía mayor cobertura (recall), pero generaba más falsos positivos, mientras que Random Forest brindó alta precisión con menor carga operativa para los equipos de calidad. La elección dependerá del enfoque estratégico que la organización desee adoptar.

Finalmente, este proyecto aporta una solución replicable, escalable y alineada con los principios de transformación digital, que puede ser aplicada en otras unidades de negocio o en empresas con estructuras operativas similares.

Recomendaciones

Se recomienda capacitar a los agentes de servicio enfocándose especialmente en las categorías donde se concentran los errores críticos, con énfasis particular en “gestión del proceso”, ya que esta tiene un alto peso en el modelo de predicción. Es fundamental integrar el modelo predictivo en dashboards interactivos como MODE, lo cual permitirá una visualización clara y accesible para los equipos operativos.

Asimismo, es conveniente automatizar alertas que identifiquen comportamientos anómalos en tiempo real, facilitando la toma de decisiones proactiva. Para asegurar la efectividad continua del modelo, se debe evaluar y actualizar periódicamente su desempeño. A esto se debe sumar un sistema continuo de retroalimentación que permita medir su impacto sobre los indicadores clave de desempeño (KPIs).

Por último, es clave realizar sesiones de calibración entre revisores para garantizar la consistencia en la evaluación de calidad. También se sugiere analizar las diferencias de desempeño entre workspaces, con el fin de diseñar intervenciones diferenciadas que respondan a las particularidades de cada operación.

Referencias Bibliográficas

- Amiri, A., Naeem, M. A., & Safi, R. (2016). Hybrid machine learning model for predicting support failures. *Knowledge-Based Systems, 98*, 224–234.
- Cao, L., & Yang, J. (2021). *Classifying customer satisfaction from support interactions using decision trees and SVM. IEEE Access, 9*, 44211–44220.
- Chatterjee, A., Sharma, S., & Bhattacharya, S. (2022). A random forest approach to sentiment-based classification in customer support systems. *Expert Systems with Applications, 198*, 116842.
- Chen, Y., & Huang, H. (2018). Predicting call center service quality using data mining techniques. *Expert Systems with Applications, 106*, 140–151.
- Farzindar, A., & Inkpen, D. (2019). Natural language processing for social media: Second edition. *Natural Language Engineering, 25*(5), 665–672.
- Kuo, Y. F., Chen, M. F., & Ting, C. Y. (2014). A text mining-based approach to evaluating service performance in financial call centers. *Journal of Intelligent Information Systems, 42*(3), 591–602.
- Mehdiyev, N., Krumeich, J., Enke, D., & Thomas, O. (2020). A hybrid machine learning approach for business process anomaly detection. *Decision Support Systems, 132*, 113248.
- Singh, R., Prasad, R., & Jha, R. (2020). Text mining-based classification of customer issues in telecom support tickets. *Journal of Big Data, 7*(1), 92.
- Xu, Y., Liu, T., & Li, J. (2017). Deep learning for customer dissatisfaction prediction in textual conversations. *Information Sciences, 385–386*, 95–103.
- Zhang, L., & Zheng, H. (2015). Predicting system failures in service quality using binary

classifiers. *Computers in Industry*, 70, 17–25.

Apéndices

Apéndice A

Código en Python Usado para Entrenamiento del Modelo

```
# Importación de librerías necesarias

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.model_selection import train_test_split

from sklearn.preprocessing import OneHotEncoder

from sklearn.linear_model import LogisticRegression

from sklearn.ensemble import RandomForestClassifier

from sklearn.metrics import classification_report, confusion_matrix,
roc_auc_score

import xgboost as xgb

# Carga ed datos

file_path = '/content/dataset_klaus.csv' # Reemplaza con la ruta de
tu archivo

df_raw = pd.read_csv(file_path)

print(f"Dataset cargado exitosamente con {df_raw.shape[0]} filas y
{df_raw.shape[1]} columnas.")

print("-" * 60)

# Verificar el porcentaje de valores nulos por columna

print("\nPorcentaje de valores nulos por columna:")

null_percentages = (df_raw.isnull().sum() / len(df_raw)) * 100
```

```
print(null_percentages.sort_values(ascending=False))
```

```
# FASE 2: PREPROCESAMIENTO Y LIMPIEZA DE DATOS

#
=====
====

print("\n--- [FASE 3.1]: Iniciando limpieza y preprocesamiento de
datos... ---")

# AJUSTE CLAVE: Se añaden 'SCORE' y 'CAUSE' a la lista de columnas a
eliminar

cols_to_drop = [
    'TICKET_ID', 'TICKET_STATUS', 'AGENT_EMAIL', 'AGENT_NAME',
    'TICKET_CREATED_AT_ZENDESK', 'ASSIGNEE_ID', 'REVIEW_ID',
    'EXTERNAL_ID', 'COMMENT'
]

# Eliminamos solo las columnas que existen en el DataFrame para
evitar errores

cols_existentes_para_eliminar = [col for col in cols_to_drop if col
in df_raw.columns]

df_clean = df_raw.drop(columns=cols_existentes_para_eliminar)

print(f"Se eliminaron {len(cols_existentes_para_eliminar)} columnas.
Columnas restantes: {df_clean.shape[1]}")

# Primero, verificamos si las columnas necesarias existen

if 'SCORE' in df_clean.columns and 'CAUSE' in df_clean.columns:

    # La modificación:
```

```
# Si SCORE es 100, se establece CAUSE como "Positive
Interaction"

condition = df_clean['SCORE'] == 100

# la asignación con .loc, usualmente no es necesario si df_clean
es el DataFrame principal.

df_clean.loc[condition, 'CAUSE'] = "Positive Interaction"

print(f"\nModificación realizada. Se actualizaron
{condition.sum()} filas en CAUSE a 'Positive Interaction'.")

# Esta parte debe estar indentada DENTRO del bloque 'if'
print("\nEjemplos DESPUÉS de la modificación (donde SCORE ==
100):")

print(df_clean[df_clean['SCORE'] == 100][['SCORE',
'CAUSE']].head())

# También es buena idea verificar si la nueva categoría aparece
en RATING_CAUSE

# Esta parte también debe estar indentada DENTRO del bloque 'if'
print("\nConteo de valores en CAUSE después de la
modificación:")

print(df_clean['CAUSE'].value_counts(dropna=False).head())

else:

# Este 'else' corresponde al 'if' que verifica la existencia de
las columnas
```

```
# y debe estar al mismo nivel de indentación que ese 'if'.

print("No se pudo realizar la modificación porque una o ambas
columnas ('SCORE', 'CAUSE') no existen en el DataFrame.")

# --- 2.2 Estandarización de la Variable Objetivo 'CRITICAL' 'CAUSE'
---

# Imputar 'CAUSE' (asumiendo que es la columna a la que te referías
como ROOT_CAUSE)

if 'CAUSE' in df_clean.columns:

    df_clean['CAUSE'] = df_clean['CAUSE'].fillna('Causa
Desconocida')

    print("Valores nulos en 'CAUSE' imputados con 'Causa
Desconocida'.")

# Este paso ahora se hace ANTES de eliminar SCORE y CAUSE.

if 'CRITICAL' in df_clean.columns and 'SCORE' in df_clean.columns:

    print("Estandarizando la columna 'CRITICAL'...")

    # Aplicar la regla de negocio: si SCORE es 0, CRITICAL debe ser
'yes'

    df_clean.loc[df_clean['SCORE'] == 0, 'CRITICAL'] = 'yes'

    print("Regla de negocio aplicada: SCORE de 0 se ha marcado como
error crítico.")

# Mapear 'yes'/'no' a 1/0. Se convierte a minúsculas para ser
robusto.

# .fillna(0) asegura que cualquier valor nulo o no reconocido se
trate como no crítico.
```

```
df_clean['CRITICAL'] =
df_clean['CRITICAL'].str.lower().map({'yes': 1, 'no':
0}).fillna(0).astype(int)

    print("La columna objetivo 'CRITICAL' ha sido estandarizada y
convertida a formato numérico (0/1).")

# --- 2.4 Imputación Final y Limpieza ---

# Imputar 'TICKET_GROUP'

if 'TICKET_GROUP' in df_clean.columns:

    df_clean['TICKET_GROUP'] =
df_clean['TICKET_GROUP'].fillna('Grupo Desconocido')

    print("Valores nulos en 'TICKET_GROUP' imputados con 'Grupo
Desconocido'.")

# Eliminar las pocas filas restantes con valores nulos

df_clean = df_clean.dropna()

print(f"Filas con nulos restantes eliminadas. Dimensiones finales
del dataset limpio: {df_clean.shape}")

print("-" * 60)

# --- 2.3 Eliminación de Columnas con Fuga de Datos ---

# AHORA, una vez que 'CRITICAL' está limpia, eliminamos las columnas
que causan la fuga.

cols_to_drop_leaky = ['SCORE', 'CAUSE']

cols_existentes_leaky = [col for col in cols_to_drop_leaky if col in
df_clean.columns]

df_clean = df_clean.drop(columns=cols_existentes_leaky)
```

```

    print(f"Se eliminaron {len(cols_existentes_leaky)} columnas con fuga
de datos ('SCORE', 'CAUSE').")

    print("\n--- [FASE 3]: Creando nuevas características (features)...
---")

    # --- 3.1 Extracción de Características de Fechas ---
    # Se añade CONVERSATION_CREATED_AT para un análisis más preciso.
    date_cols = ['REVIEW_CREATED_AT', 'UPDATED',
'CONVERSATION_CREATED_AT']

    for col in date_cols:

        if col in df_clean.columns:

            df_clean[col] = pd.to_datetime(df_clean[col])

            df_clean[f'{col}_mes'] = df_clean[col].dt.month

            df_clean[f'{col}_dia_semana'] = df_clean[col].dt.dayofweek #
Lunes=0, Domingo=6

            df_clean[f'{col}_hora'] = df_clean[col].dt.hour

            print(f"Características extraídas de la columna '{col}'.")

    # Eliminar las columnas de fecha originales
    df_featured = df_clean.drop(columns=date_cols)
    print("Columnas de fecha originales eliminadas.")
    print("-" * 60)
    print("\n--- [FASE 4]: Realizando Análisis Exploratorio de Datos
(EDA)... ---")

    # --- 4.1 Análisis de la Variable Objetivo ---

```

```

critical_distribution =
df_featured['CRITICAL'].value_counts(normalize=True) * 100

print("Distribución de la variable objetivo 'CRITICAL'
(corregida):")

print(critical_distribution)

if len(critical_distribution) < 2 or critical_distribution.min() <
10:

    print("\nADVERTENCIA: Las clases están extremadamente
desbalanceadas o solo existe una clase.")

# --- 4.2 Análisis Bivariado ---
# Relación entre WORKSPACE_NAME y CRITICAL

print("\n--- Tasa de Errores Críticos por Workspace ---")

workspace_critico =
df_featured.groupby('WORKSPACE_NAME')['CRITICAL'].mean().sort_values(ascen
ding=False)

display(workspace_critico.to_frame())

# Relación entre REVIEWER_NAME y CRITICAL

print("\n--- Tasa de Errores Críticos por Revisor (Top 10) ---")

revisor_critico =
df_featured.groupby('REVIEWER_NAME')['CRITICAL'].mean().sort_values(ascend
ing=False).head(10)

display(revisor_critico.to_frame())

print("-" * 60)

```

```
print("\n--- [FASE 5]: Preparando datos y entrenando modelos... ---
")

# --- 5.1 Definición de X e y ---
X = df_featured.drop('CRITICAL', axis=1)
y = df_featured['CRITICAL']

# --- 5.2 Codificación de Variables Categóricas (One-Hot Encoding) ---
--
categorical_features = X.select_dtypes(include=['object']).columns
encoder = OneHotEncoder(handle_unknown='ignore',
sparse_output=False)

# Crear un DataFrame con las columnas codificadas, asegurándose de
mantener el índice original
encoded_cols = pd.DataFrame(
    encoder.fit_transform(X[categorical_features]),
    index=X.index, # Clave: se asigna el índice original
    columns=encoder.get_feature_names_out(categorical_features)
)

# Unir las columnas codificadas al DataFrame original (que contiene
las numéricas) y eliminar las categóricas originales
X_final = X.drop(columns=categorical_features).join(encoded_cols)

print(f"Variables categóricas codificadas. El dataset final para el
modelo tiene {X_final.shape[1]} características.")
```

```

    print(f"Verificación de dimensiones: X_final tiene
    {X_final.shape[0]} filas, y tiene {y.shape[0]} filas.")

    # --- 5.3 VERIFICACIÓN DE SEGURIDAD ANTES DE LA DIVISIÓN ---
    y_counts = y.value_counts()

    print("\nVerificación de conteo de clases en la variable objetivo
    'y':")

    print(y_counts)

    # AJUSTE: Se añade una verificación explícita para ambos casos de
    error.

    if len(y_counts) < 2:

        print("\n\n!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        !!!!!!!!!!!!!")

        print("ERROR CRÍTICO: Solo se encontró una clase en la variable
        objetivo.")

        print("No es posible realizar una división estratificada para el
        entrenamiento y la prueba.")

        print("\nRECOMENDACIÓN: Revise los filtros de datos. Es posible
        que todas las muestras pertenezcan a 'No Crítico'.")

        print("!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
        !!!!!!\n")

        raise ValueError("Datos insuficientes para el modelado. Se
        necesita más de una clase.")

    elif y_counts.min() < 2:

```



```

# Modelo 1: Regresión Logística

log_reg = LogisticRegression(random_state=42,
class_weight='balanced', max_iter=1000)

log_reg.fit(X_train, y_train)

print("1. Regresión Logística entrenada.")

# Modelo 2: Random Forest

rand_forest = RandomForestClassifier(random_state=42,
class_weight='balanced', n_estimators=100)

rand_forest.fit(X_train, y_train)

print("2. Random Forest entrenado.")

# Modelo 3: XGBoost

xgb_model = xgb.XGBClassifier(random_state=42,
use_label_encoder=False, eval_metric='logloss')

xgb_model.fit(X_train, y_train)

print("3. XGBoost entrenado.")

print("-" * 60)

print("\n--- [FASE 6]: Evaluando el rendimiento de los modelos... --
-")

# --- 6.1 Generación de Reportes de Clasificación ---

models = {

```

```

    "Regresión Logística": log_reg,

    "Random Forest": rand_forest,

    "XGBoost": xgb_model

}

for name, model in models.items():

    print(f"\n--- Reporte de Clasificación para: {name} ---")

    y_pred = model.predict(X_test)

    print(classification_report(y_test, y_pred))

    # Matriz de Confusión

    cm = confusion_matrix(y_test, y_pred)

    sns.heatmap(cm, annot=True, fmt='d', cmap='Blues')

    plt.title(f'Matriz de Confusión - {name}')

    plt.ylabel('Etiqueta Real')

    plt.xlabel('Etiqueta Predicha')

    plt.show()

# --- 6.2 Interpretación del Modelo (Feature Importance) ---
print("\n--- Importancia de las Características (Random Forest) ---
")

# Usamos Random Forest porque es robusto y fácil de interpretar
feature_importances = pd.DataFrame({

    'feature': X_final.columns,

    'importance': rand_forest.feature_importances_

}).sort_values('importance', ascending=False).head(20) # Top 20

display(feature_importances)

```

```
plt.figure(figsize=(10, 8))
sns.barplot(x='importance', y='feature', data=feature_importances)
plt.title('Top 20 Características más Importantes (Random Forest)')
plt.show()

print("-" * 60)

print("\n--- PROCESO COMPLETADO ---")

# Define the date columns
date_cols = ['REVIEW_CREATED_AT', 'UPDATED',
'CONVERSATION_CREATED_AT']

# Convert date columns to datetime objects
for col in date_cols:
    if col in df_raw.columns:
        df_raw[col] = pd.to_datetime(df_raw[col], errors='coerce') #
Use errors='coerce' to handle potential parsing issues

# Find the earliest and latest dates in the original date columns
earliest_date = df_raw[date_cols].min().min()
latest_date = df_raw[date_cols].max().max()

print(f"El rango de fechas del dataset es desde
{earliest_date.strftime('%Y-%m-%d %H:%M:%S')} hasta
{latest_date.strftime('%Y-%m-%d %H:%M:%S')}")
```

Apéndice B

Autorización Empresa VI

V0.1

Bogotá, 20 de enero de 2025

Señor:

LAURA BRAVO

Gerente de Calidad y Entrenamiento

Asunto: Autorización para la ejecución del proyecto titulado: *Modelo predictivo para optimizar la calidad del servicio en PayJoy Inc.*

Cordial saludo estimado Gerente,

Como es de su conocimiento, actualmente me encuentro adelantando estudios de posgrado en la Especialización en Analítica de Datos ofertado por la Universidad Nacional Abierta y a Distancia "UNAD". Para finalizar mi proceso académico es mi objetivo desarrollar un trabajo de grado aplicado a PayJoy Inc, de manera que pueda aportar mis conocimientos adquiridos y generar un impacto positivo en la empresa, relacionado con los temas de Calidad del servicio, Servicio al cliente, Machine Learning, Análisis de datos, Optimización operativa, motivo por el cual, muy comedidamente solicito su autorización y aprobación para la ejecución del proyecto titulado: *Modelo predictivo para optimizar la calidad del servicio en PayJoy Inc* el cual se encuentra avalado por parte la Institución de educación superior "UNAD".

El proyecto en su objetivo general describe lo siguiente: "ingresar objetivo general"; al mismo tiempo será apoyado por los objetivos específicos: "ingresar objetivos específicos" para obtener como resultado un alto impacto en la organización *PayJoy Inc*.

De obtener esta autorización, se elaborará un acuerdo de confidencialidad para proteger la identidad de la empresa y sus activos de información; a su vez se destacan los siguientes procesos para ser garantes en la transparencia de la ejecución del proyecto:

- Se prohíbe la ejecución de cualquier tipo de pruebas de seguridad que no estén autorizadas expresamente por PayJoy Inc (Aplica para la especialización en Seguridad Informática).
- La empresa *PayJoy Inc* deberá establecer qué tipo de información es privada y cuál es pública para delimitar el acceso y aplicación de pruebas en la ejecución del proyecto.
- La solicitud de información al igual que ejecución de pruebas deben quedar por escrito y se genera un informe de resultados periódicamente el cual será compartido con el gerente de la organización o empresa.
- La persona autorizada siempre debe operar dentro de la ley 1273 de 2009 y de las demás regulaciones establecidas en la empresa.
- Respetar la privacidad de todos los individuos y mantener su privacidad en los reportes. Está prohibida divulgar información personal en tales reportes.

El resultado del proyecto se verá reflejado en un documento el cual será cargado al repositorio institucional de la Universidad Nacional Abierta y a Distancia "UNAD". El documento ampara la confidencialidad y anonimato de la empresa, estos aspectos se encuentran estipulados en el acuerdo de confidencialidad; agradezco el apoyo prestado en esta etapa de mi carrera profesional.

Firman en Bogotá., a los (20) días del mes de (enero) de 2025

Apéndice C

Acuerdo de Confidencialidad - Empresa Estudiante VI



V 0.1

ACUERDO DE CONFIDENCIALIDAD ENTRE MARIA AMAYA – LUIS MORANTES Y PAYJOY COLOMBIA

Por la parte reveladora

Nombre: PayJoy Inc
Dirección: Calle 90 # 12^a
Teléfono: (601) 48xxx66
E-mail: clientesxxxx@paxxxx.com

Por la parte receptora de la información

Nombre: Maria Amaya / Luis Morantes
Dirección: Calle 90 # 12^a
Teléfono: 31xxxxx87 / 3xxxxx9
E-mail: maxxxxx@xxx.com / luixxxx@xxx.com

Identificación del proyecto

Entre los firmantes, identificados anteriormente, hemos convenido en celebrar el presente acuerdo de confidencialidad previa las siguientes **CONSIDERACIONES**

1. Que la información compartida en virtud del presente acuerdo pertenece a la *PayJoy Inc*, y la misma es considerada sensible y de carácter restringido en su divulgación, manejo y utilización. Dicha información es compartida en virtud del desarrollo del *Proyecto aplicado con el título: Modelo predictivo para optimizar la calidad del servicio en PayJoy Inc.*



V 0.1

2. Que la información de propiedad de *PayJoy Inc* ha sido desarrollada u obtenida legalmente, como resultado de sus procesos, programas o proyectos y, en consecuencias abarca documentos, datos, tecnología y/o material que considera único y confidencial, o que es objeto de protección a título de secreto industrial.
3. Que el presente acuerdo se realiza por un lado entre la parte receptora de la información como integrante del proyecto de *Modelo predictivo para optimizar la calidad del servicio en PayJoy Inc, Maria Amaya / Luis Morantes* que, para el presente caso actual como **revelador, guarda y administrador** de la información de propiedad de *PayJoy Inc*.

En consecuencia, **las partes** se suscriben a las siguientes cláusulas:

Primera. Objeto: en virtud del presente **acuerdo de confidencialidad**, la **parte receptora**, se obliga a no divulgar directa, indirecta, próxima a remotamente, ni a través de ninguna otra persona o de sus subalternos o funcionarios, asesores o cualquier persona relacionada con ella, la **información confidencial** perteneciente al *PayJoy Inc*, así como también a no utilizar dicha información en beneficio propio ni de terceros, sólo con fines estadísticos y de mejoramiento de la *PayJoy Inc*.

Segunda. Definición de información confidencial: se entiende como **Información Confidencial**, para los efectos del presente acuerdo:

1. La información que no sea pública y sea conocida por la **parte receptora** con ocasión del proyecto en desarrollo.
2. Cualquier información societaria, técnica, jurídica, financiera, comercial, de mercado, estratégica, de productos, nuevas tecnologías, patentes, modelos de utilidad, diseños industriales, modelos de negocios, información del personal de la

organización, información relacionada con la estructura organizacional y/o cualquier otra relacionada con el proyecto *Modelo predictivo para optimizar la calidad del servicio en PayJoy Inc*, bien sea escrita, oral o visual, o en cualquier forma tangible o no, incluidos los mensajes de datos (en la forma definida en la ley), de la cual, la **parte receptora** tenga conocimiento o a la que tenga acceso por cualquier medio o circunstancia en virtud de las reuniones sostenidas y/o documentos suministrados.

3. La que corresponda o deba considerarse como tal para garantizar el derecho constitucional a la intimidad, la honra y el buen nombre de las personas y deba guardarse la debida diligencia en su discreción y manejo en el desempeño de sus funciones.

Tercera. Origen de la información confidencial: provendrá de documentos suministrados en el desarrollo del proyecto y que tiene que ver con las creaciones del intelecto, a la naturaleza, medios, formas de distribución, comercialización de productos o de prestación de servicios, transmitida verbal, visual o materialmente, por escrito en los documentos, medios electrónicos, discos ópticos, microfilmes, películas, e-mail u otros elementos similares suministrados de manera tangible o intangible, independiente de su fuente o soporte y sin que requiera advertir su carácter confidencial.

Cuarta. Obligaciones de la parte receptora: Se considerará como **parte receptora** de la **información confidencial** a la persona que recibe la información, o que tenga acceso a ella. La parte receptora se obliga a:

1. Mantener la **información confidencial** segura, usarla solamente para los propósitos relacionados con él, en caso de ser solicitada, devolverla toda (incluyendo copias de esta) en el momento en que ya no requiera hacer uso de la misma o cuando termine la relación, caso en el cual, deberá entregar dicha información antes de la terminación de la vinculación.



V 0.1

2. Proteger la **información confidencial**, sea verbal, escrita, visual, tangible, intangible o que por cualquier otro medio reciba, siendo legítima poseedora de la misma *PayJoy Inc*, restringiendo su uso exclusivamente a las personas que tengan absoluta necesidad de conocerla.
3. Abstenerse de publicar la **información confidencial** que conozca, reciba o intercambie con ocasión de las reuniones sostenidas.
4. Usar la **información confidencial** que se le entregue, únicamente para los efectos señalados al momento de la entrega de dicha información.
5. Mantener la **información confidencial** en reserva hasta tanto adquiera el carácter de pública.
6. Responder por el mal uso que le den sus representantes a la **información confidencial**.
7. Guardar la reserva de la **información confidencial** como mínimo, con el mismo cuidado con la que protege la **información confidencial**.
8. La **parte receptora** se obliga a no transmitir, comunicar revelar o de cualquier otra forma divulgar total o parcialmente, pública o privadamente, la **información confidencial** sin el previo consentimiento por escrito por parte de *PayJoy Inc*.
9. La **parte receptora** se compromete a establecer que los datos a utilizar son: Se tomara la base de datos anteriormente usada para el Form Q&A Colombia, de la cual haremos cambio de datos sensibles, como lo son nombres, apellidos y de haberlos números de teléfono, correos o cualquier tipo de dato que no esté amparado por la Ley 1581 de 2012, esto con el fin de mantener la confidencialidad de la información ya que se tendrá que realizar



V 0.1

entregas a la universidad sobre el funcionamiento del modelo predictivo a realizar con dicha base.

10. La información capturada por la **parte receptora** se observará como *cifras para estudio estadístico, comparativo, información cualitativa*, no existirá ningún tipo de ganancia económica, es netamente con fines educativos.
11. La identidad de todo el personal de **PayJoy Inc** no será revelada, dado que no se capturará sus nombres completos ni algún otro tipo de información que revele su identidad física o digital.
12. Las pruebas realizadas por la **parte receptora** nunca pondrán en peligro los activos tecnológicos de **PayJoy Inc**, ni violentará la ley de delitos informáticos Colombiana 1273 de 2009 estando en el margen de las buenas prácticas y los procesos legales pertinentes.
13. El estudiante *Maria Amaya / Luis Morantes* se compromete a difuminar, bloquear y ocultar toda información que revele la identidad de la empresa *PayJoy Inc* para salvaguardar la confidencialidad e identidad de la misma en el documento final del proyecto el cual será publicado en el repositorio institucional y de acceso público.
14. La empresa **PayJoy Inc**, autorizara, SI su nombre aparecerá en el título del proyecto, dado el caso, de que no sea autorizado, este deberá adecuarse.

Parágrafo: Cualquier divulgación autorizada de la **información confidencial** a terceras personas estará sujeta a las mismas obligaciones de confidencialidad derivadas del presente **Acuerdo** y la **parte receptora** deberá informar estas restricciones incluyendo la identificación de la información como confidencial.



V 0.1

Quinta. Obligaciones de la parte reveladora: Son obligaciones de la parte reveladora:

1. Mantener la reserva de la **información confidencial** hasta tanto adquiera el carácter de pública.
2. Documentar toda la **información confidencial que** transmita de manera escrita, oral o visual, mediante documentos, medios electrónicos, discos ópticos, microfilmes, películas, e-mails u otros elementos similares o en cualquier forma tangible o no, incluidos los mensajes de datos, como registro de la misma para la determinación de su alcance, e indicar específicamente y de manera clara e inequívoca el carácter confidencial de la información suministrada de la **parte receptora**.

Sexta. Exclusiones a la confidencialidad: La **parte receptora** queda relevada o eximida de la obligación de confidencialidad, únicamente en los siguientes casos:

1. Cuando la **información confidencial** haya sido o sea de dominio público. Si la información se hace de dominio público durante el plazo del presente acuerdo, por un hecho ajeno a la **parte receptora**, esta conservará su deber de reserva sobre la información que no haya sido afectada.
2. Cuando la **información confidencial** deba ser revelada por sentencia en firme de un tribunal o autoridades competentes en desarrollo de sus funciones que ordenen el levantamiento de la reserva y soliciten el suministro de esta información. No obstante, en este caso la parte reveladora será la encargada de dar cumplimiento a la orden, restringiendo la divulgación a la información estrictamente necesaria, y en el evento de que la confidencialidad se mantenga, no eximirá a la parte receptora del deber de reserva.



V 0.1

3. Cuando la **parte receptora pruebe** que la **información confidencial** ha sido obtenida por otras fuentes.
4. Cuando la **información confidencial** ya la tenía en su poder la parte receptora antes de la entrega de la información reservada.

Séptima. Responsabilidad: la parte que contravenga el acuerdo será responsable ante la otra parte o ante los terceros de buena fe sobre los cuales se demuestre que se han visto afectados por la inobservancia del presente **acuerdo**, por los perjuicios morales y económicos que estos puedan sufrir como resultado del incumplimiento de las obligaciones aquí contenidas.

Octava. Solución de controversias: Las partes *Maria Amaya / Luis Morantes – PayJoy Inc*) se comprometen a esforzarse en resolver mediante los mecanismos alternativos de solución de conflictos cualquier diferencia que surja con motivo de la ejecución del presente **acuerdo**. En caso de no llegar a una solución directa para la controversia planteada, someterán la cuestión controvertida a las leyes colombianas y a la jurisdicción competente en el momento de presentarse la diferencia. La Universidad Nacional Abierta y a Distancia como institución educativa no se hace responsable del no cumplimiento de las cláusulas del presente acuerdo de confidencialidad por parte de *Maria Amaya / Luis Morantes*.

Novena. Legislación aplicable: Este **acuerdo** se regirá por las leyes de la República de Colombia y se interpretará de acuerdo con las mismas.



V 0.1

Décima. Aceptación del Acuerdo: Las partes han leído y estudiado de manera detenida los términos y el contenido del presente **Acuerdo** y por tanto manifiestan estar conformes y aceptan todas las condiciones.

Firman en Bogotá., a los (20) días del mes de (Enero) de 2025

Como Parte Receptora:

Por la parte reveladora:

Nombre del estudiante.
Estudiante UNAD.
C.C. No. 1.0xxxx.611

Gerente de Calidad
PayJoy Inc