

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

FACULTAD DE INGENIERÍA ELÉCTRICA, ELECTRÓNICA,
INFORMÁTICA Y MECÁNICA

ESCUELA PROFESIONAL DE INGENIERÍA INFORMÁTICA Y DE
SISTEMAS



TESIS

**SISTEMA DE GESTIÓN E INTEGRACIÓN DE
INFORMACIÓN CONTABLE Y FINANCIERA BASADO EN
RPA, ETL Y AWS PARA LA EMPRESA TOPITOP**

PRESENTADO POR:

Br. LETICIA GIULIANA OLAZABAL CALLER

Br. DIEGO MARTIN SALAZAR ORE

**PARA OPTAR AL TÍTULO PROFESIONAL DE
INGENIERO INFORMÁTICO Y DE SISTEMAS**

ASESOR:

Dr. DENNIS IVAN CANDIA OVIEDO

CUSCO - PERÚ

2026



Universidad Nacional de San Antonio Abad del Cusco

INFORME DE SIMILITUD

(Aprobado por Resolución Nro.CU-321-2025-UNSAAC)

El que suscribe, el **Asesor** DENNIS IVAN CANDIA OVIEDO.....
..... quien aplica el software de detección de similitud al
trabajo de investigación/tesis titulada: SISTEMA DE GESTION E INTEGRACION
DE INFORMACION CONTABLE Y FINANCIERA BASADO EN RPA, ETL
Y AWS PARA LA EMPRESA TOPITOP

Presentado por: LETICIA GIULIANA OLAZABAL CALLER DNI N° 73145986 ;
presentado por: DIEGO MARTIN SALAZAR ORE DNI N°: 70128519

Para optar el título Profesional/Grado Académico de INGENIERO INFORMATICO
Y DE SISTEMAS

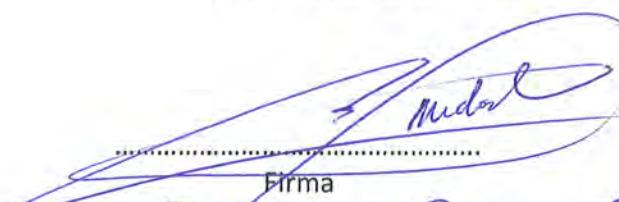
Informo que el trabajo de investigación ha sido sometido a revisión por 2 veces, mediante el
Software de Similitud, conforme al Art. 6° del **Reglamento para Uso del Sistema Detección de
Similitud en la UNSAAC** y de la evaluación de originalidad se tiene un porcentaje de 9 %.

**Evaluación y acciones del reporte de coincidencia para trabajos de investigación conducentes a
grado académico o título profesional, tesis**

Porcentaje	Evaluación y Acciones	Marque con una (X)
Del 1 al 10%	No sobrepasa el porcentaje aceptado de similitud.	<input checked="" type="checkbox"/>
Del 11 al 30 %	Devolver al usuario para las subsanaciones.	
Mayor a 31%	El responsable de la revisión del documento emite un informe al inmediato jerárquico, conforme al reglamento, quien a su vez eleva el informe al Vicerrectorado de Investigación para que tome las acciones correspondientes; Sin perjuicio de las sanciones administrativas que correspondan de acuerdo a Ley.	

Por tanto, en mi condición de Asesor, firmo el presente informe en señal de conformidad y **adjunto**
las primeras páginas del reporte del Sistema de Detección de Similitud.

Cusco, 16 de ENERO de 2026


Firma
Post firma Dennis I. Candia Oviedo
Nro. de DNI. 23987482

ORCID del Asesor 0000-0001-6673-6174

Se adjunta:

1. Reporte generado por el Sistema Antiplagio.
2. Enlace del Reporte Generado por el Sistema de Detección de Similitud: **oid: 27259:546015398**

DENNIS IVAN CANDIA OVIEDO

TESIS_FINAL OLAZABAL & SALAZAR.pdf

 Universidad Nacional San Antonio Abad del Cusco

Detalles del documento

Identificador de la entrega

trn:oid:::27259:546815398

133 páginas

Fecha de entrega

16 ene 2026, 1:22 p.m. GMT-5

20.083 palabras

Fecha de descarga

26 ene 2026, 8:02 a.m. GMT-5

115.042 caracteres

Nombre del archivo

TESIS_FINAL OLAZABAL & SALAZAR.pdf

Tamaño del archivo

10.7 MB

9% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Trabajos entregados

Exclusiones

- N.º de fuente excluida
- N.º de coincidencias excluidas

Fuentes principales

9%	 Fuentes de Internet
2%	 Publicaciones
0%	 Trabajos entregados (trabajos del estudiante)

Marcas de integridad

N.º de alerta de integridad para revisión

Caracteres reemplazados

41 caracteres sospechosos en N.º de páginas

Las letras son intercambiadas por caracteres similares de otro alfabeto.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

Dedicatorias

A Dios por guiarme.

A mi padre, por ser mi mayor inspiración. Su esfuerzo y sus enseñanzas siguen presentes en cada paso que doy, aunque ya no esté a mi lado, sigue siendo mi guía.

A mi madre, por ser mi mayor ejemplo de fuerza y perseverancia; su apoyo hizo posible este logro.

A mi hermano, por su compañía constante y ánimo.

Leticia

A Dios por darme la fortaleza y la sabiduría necesarias para culminar esta etapa.

A mis padres, por su amor y apoyo incondicional a lo largo de mi formación.

A mi hermano y hermanas, por su compañía constante, y a mis sobrinos, por ser una fuente de alegría y motivación en este camino.

Diego

Agradecimientos

Gracias a Dios, mi familia, mis docentes y a todos los que me apoyaron durante este proyecto.

Leticia

Gracias a Dios, mis padres, hermanos, sobrinos, docentes y a todos los que me apoyaron para culminar esta etapa.

Diego

Resumen

La empresa Topitop enfrenta dificultades en la gestión contable y financiera debido a la descentralización de la información en diversos sistemas internos y externos. Esta dispersión complica el proceso de conciliación o matching de documentos contables entre sus distintas plataformas. Para resolver este problema, se ha implementado un sistema automático de integración y procesamiento de información, utilizando Robotic Process Automation (RPA), procesos de ETL (Extract, Transform, Load) y servicios en la nube de Amazon Web Services (AWS). La solución contempla la automatización de tareas rutinarias relacionadas con la generación de reportes de documentos contables mediante RPA, así como la extracción y transformación de datos provenientes del sistema RMS (Retail Management System), el portal web de ACEPTA y el sistema SAP. Todos estos datos se integran en una estructura común para su análisis. El sistema se despliega sobre AWS, permitiendo alta disponibilidad, escalabilidad y seguridad en el procesamiento de datos. Esta propuesta elimina errores operativos y permite a los usuarios de las áreas involucradas identificar de inmediato el documento con error, conocer en qué sistema falta y tomar acción de forma oportuna. Así, la empresa no solo mejora la eficiencia y la toma de decisiones, sino que también logra que toda la información esté alineada y disponible.

Palabras clave: Robotic process automation, Amazon web services, Retail management system, ETL.

Abstract

Topitop faces challenges in accounting and financial management due to the decentralization of information across various internal and external systems. This dispersion complicates the process of reconciliation or matching of accounting documents among its different platforms. To address this problem, an automated system for information integration and processing has been implemented, using Robotic Process Automation (RPA), Extract, Transform, Load (ETL) processes, and Amazon Web Services (AWS) cloud services. The solution includes the automation of routine tasks related to the generation of accounting document reports through RPA, as well as the extraction and transformation of data from the RMS (Retail Management System), the ACEPTA web portal, and the SAP system. All these data are integrated into a common structure for analysis. The system is deployed on AWS, enabling high availability, scalability, and security in data processing. This proposal eliminates operational errors and allows users from the involved areas to identify the document with an error, determine in which system it is missing, and take immediate action. Thus, the company not only improves efficiency and decision-making but also ensures that all information is aligned and readily available.

Keywords: Robotic process automation, Amazon web services, Retail management system, ETL.

Introducción

En la actualidad, el entorno empresarial es dinámico y competitivo, por lo que las organizaciones deben apoyarse en la tecnología para optimizar sus procesos internos y responder con eficiencia a las exigencias del mercado. En este contexto, una adecuada gestión de la información contable y financiera resulta clave para asegurar transparencia, control y una toma de decisiones oportuna.

Topitop, empresa del sector retail a nivel nacional, enfrenta dificultades asociadas a la dispersión de datos y al *matching* manual de información contable proveniente de sistemas como el Retail Management System (RMS), el portal web de ACEPTA y el sistema SAP. Esta situación evidencia la necesidad de una solución que centralice y automatice el tratamiento de la información.

Por ello, esta tesis propone el desarrollo de un Sistema de Gestión e Integración de Información Contable y Financiera basado en RPA (Robotic Process Automation), procesos ETL (Extract, Transform and Load) y servicios en la nube de Amazon Web Services (AWS). La solución busca automatizar tareas repetitivas, reducir errores humanos y consolidar información en un entorno único, accesible y confiable.

Finalmente, se presenta el diseño e implementación del sistema y su impacto en la organización, con el propósito de servir como referencia para futuras iniciativas de transformación digital en el sector retail y en contextos con necesidades similares de integración y automatización de datos.

Abreviaturas

API: Interfaz de Programación de Aplicaciones (Application Programming Interface)

AWS: Amazon Web Services

BPMN: Business Process Model and Notation

CDC: Captura de Datos de Cambio (Change Data Capture)

DOD-ETL: Extracción, Transformación y Carga Bajo Demanda (Distributed On-Demand ETL)

ETL: Extracción, Transformación y Carga (Extract, Transform, Load)

IP: Protocolo de Internet (Internet Protocol)

KPI: Indicador Clave de Desempeño (Key Performance Indicator)

PSE: Proveedor de Servicios Electrónicos

RMS: Sistema de Gestión de Ventas Minoristas (Retail Management System)

RPA: Automatización Robótica de Procesos (Robotic Process Automation)

SAP: Sistemas, Aplicaciones y Productos en Procesamiento de Datos

SP: Procedimiento almacenado (Store Procedure)

SUNAT: Superintendencia Nacional de Administración Tributaria

Índice general

Dedicatorias	II
Agradecimientos	III
Resumen	IV
Abstract	V
Introducción	VI
Abreviaturas	VII
Índice de figuras	XII
Índice de tablas	XV
1. Aspectos Generales	1
1.1. Descripción de la empresa	1
1.2. Organigrama de la empresa	2
1.3. Planteamiento del problema	3
1.3.1. Descripción del problema	3
1.3.2. Identificación del problema	5
1.4. Formulación del problema	5
1.4.1. Problema General	5
1.4.2. Problemas Específicos	6
1.5. Objetivos	6

ÍNDICE GENERAL

1.5.1. Objetivo General	6
1.5.2. Objetivos Específicos	6
1.6. Justificación	7
1.7. Presupuesto	8
1.8. Contribuciones	8
1.9. Impacto Social	8
2. Metodología	10
2.1. Alcance	10
2.2. Diseño metodológico	11
2.2.1. Diseño de la investigación	11
2.2.2. Tipo de la investigación	11
2.2.3. Método de investigación	11
2.3. Para el desarrollo del sistema	12
3. Marco Teórico	14
3.1. Antecedentes	14
3.1.1. Antecedentes Internacionales	14
3.1.2. Antecedentes Nacionales	16
3.2. Bases Teóricas	20
3.2.1. Sistema de Gestión Contable	20
3.2.2. Conciliación Bancaria	21
3.2.3. Sistema de Gestión Financiera	21
3.2.4. Gestión de tesorería	22
3.2.5. Cumplimiento fiscal y regulatorio	22
3.2.6. ACEPTA	23
3.2.7. Scrum	24
3.2.8. Automatización Robótica de Procesos (RPA)	33

ÍNDICE GENERAL

3.2.9. Extracción, transformación y carga (ETL)	35
3.2.10. Amazon Web Services (AWS)	39
3.2.11. Pandas	42
3.2.12. Selenium	42
4. Desarrollo del sistema	44
4.1. Metodología del proyecto	44
4.1.1. Aplicación de Scrum	44
4.2. Historias de usuario por Sprint	45
4.3. Automatización de descarga de documentos del portal ACEPTA usando RPA.	51
4.4. Extracción y transformación de datos de los documentos del portal ACEPTA. .	63
4.5. Extracción y transformación de datos de los documentos de la Base de datos del sistema Retail Management System.	65
4.6. Extracción y transformación de datos de los documentos de la Base de datos de SAP.	77
4.7. Sistematización e integración de la información de los tres sistemas. . .	80
4.8. Generación de reportes y/o alerta.	82
4.9. Despliegue en Amazon Web Services (AWS).	84
4.9.1. Diagrama de despliegue	84
4.9.2. Crear una instancia EC2	86
4.9.3. Conectarse a la Instancia	89
4.9.4. Preparar el servidor	90
4.9.5. Clonar el repositorio	97
4.9.6. Automatizar con Cron	102
5. Análisis y discusión de resultados	105
5.1. Resultados	105
5.2. Verificación del comportamiento del sistema	106
5.3. Análisis de resultados respecto a los objetivos	108

ÍNDICE GENERAL

Conclusiones	112
Recomendaciones	113
Bibliografía	114

Índice de figuras

1.	Organigrama de la empresa Topitop.	2
2.	Proceso manual de los colaboradores	4
3.	Scrum framework	25
4.	Proceso de extracción, transformación y carga (ETL)	36
5.	Portal de Acepta	51
6.	Login de Acepta	52
7.	Menú emitidos	54
8.	Búsqueda avanzada en documentos emitidos	54
9.	Menús desplegables en documentos emitidos	56
10.	Exportar documentos emitidos	57
11.	Modal de confirmación de la exportación	58
12.	Ir al menú de reportes	59
13.	Espera de la habilitación del botón de descarga	61
14.	Botón habilitado de descarga	61
15.	Diagrama de base de datos que utiliza el SP	67

ÍNDICE DE FIGURAS

16. Diagrama de despliegue	85
17. E2C Instancias	86
18. Seleccionar instancia de maquina virtual	87
19. Seleccionar tipo de instancia	87
20. Habilitar acceso SSH para la instancia	88
21. Crear instancia	88
22. Conexión a la maquina virtual	89
23. Actualización de paquetes	90
24. Instalación de Python por terminal	91
25. Habilitación de módulos adicionales en Amazon	92
26. Habilitación de módulos concluida	92
27. Descarga de Google Chrome con wget desde el sitio oficial	93
28. Instalación local de Google Chrome con yum	94
29. Instalación local de Google Chrome en proceso	94
30. Instalación local de Google Chrome concluida	95
31. Descarga del archivo ChromeDriver desde el repositorio oficial de Google para pruebas automatizadas.	96
32. Verificación de la versión instalada de ChromeDriver.	97
33. Instalación de Git en Linux usando yum.	98
34. Instalación de Git culminada	98

ÍNDICE DE FIGURAS

35. Clonación del repositorio de la tesis desde GitHub mediante git clone.	99
36. Ingreso al directorio del proyecto clonado.	100
37. Creación y activación del entorno virtual en el proyecto.	100
38. Instalación de dependencias del proyecto con pip.	101
39. Instalación de dependencias del proyecto con pip culminada.	101
40. Instalación del servicio de tareas programadas cronie en Linux.	102
41. Activación y verificación del servicio crond para tareas programadas. . . .	103
42. Edición del archivo crontab para programar tareas automáticas.	103
43. Guardar el archivo crontab	104
44. Verificación de tareas programadas con crontab -l.	104
45. Captura de Documentos Procesados en ACEPTA y no en RMS.	106
46. Captura de Documentos Procesados en RMS y no en ACEPTA.	107
47. Captura de Documentos Procesados en SAP y no en RMS.	107
48. Captura de Documentos Procesados en SAP y no en ACEPTA.	108
49. Procesos automatizados	110

Índice de tablas

1.	Detalle del tiempo invertido en los procedimientos manuales	4
2.	Presupuesto para el desarrollo del proyecto	8
3.	Requerimientos funcionales .	46
4.	Historias de usuario, backlog y relación con requerimientos y sprints . . .	50
5.	Detalle del tiempo invertido en los procedimientos automatizados	110
6.	Resumen comparativo antes vs. después con ahorro absoluto y relativo .	111

Capítulo 1

Aspectos Generales

1.1. Descripción de la empresa

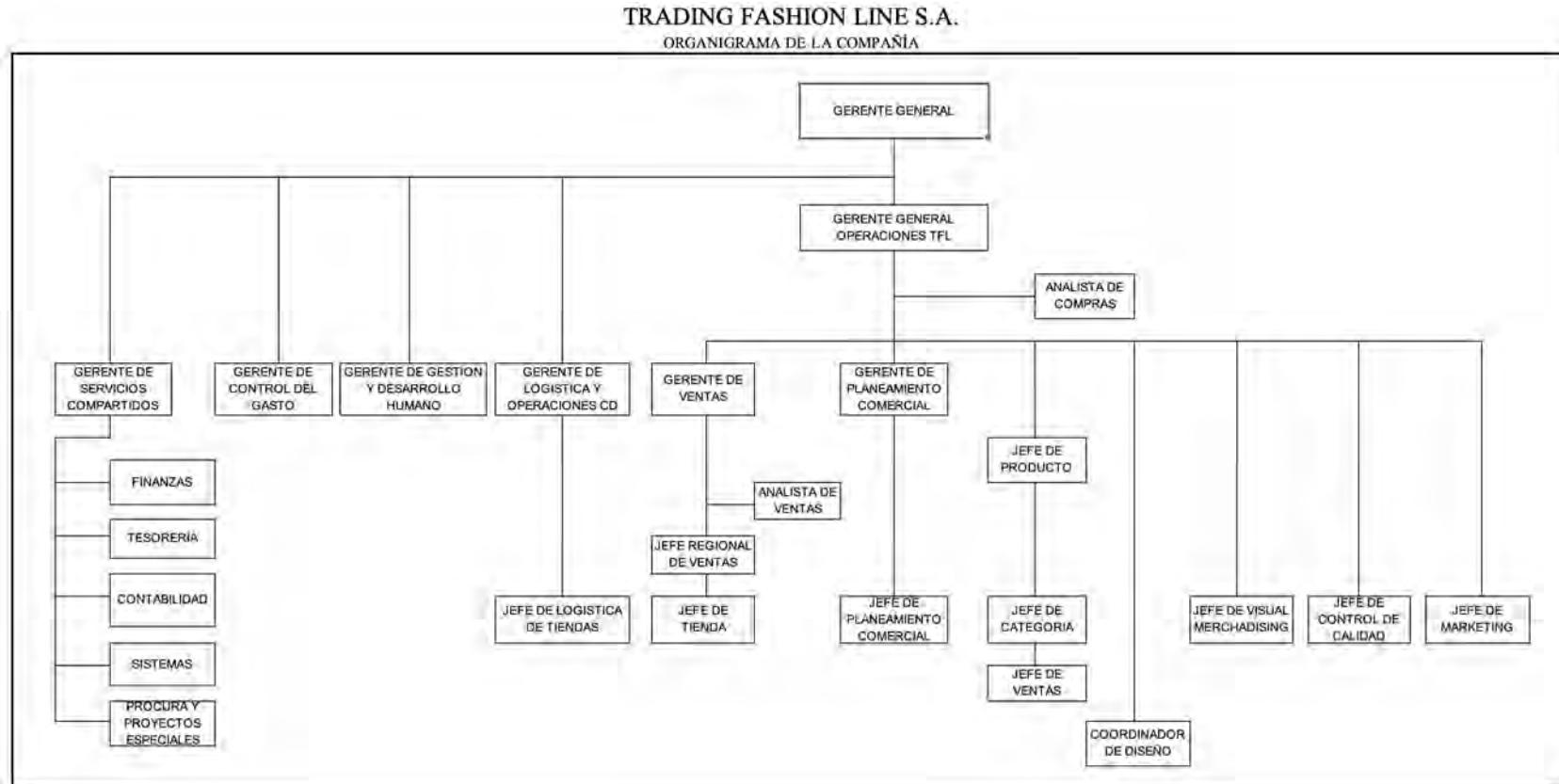
Topitop es una reconocida cadena de tiendas de ropa con sede en Perú, que exporta el 70 % de su producción, siendo los principales destinos: EEUU y Alemania. La otra parte se distribuye a través de la cadena de tiendas Topitop ubicadas en Perú, Venezuela y Ecuador. Cuenta con 66 tiendas a nivel nacional en Perú y con unas ventas anuales de alrededor de 280 millones de soles, se destaca por ofrecer una amplia gama de prendas de vestir para toda la familia, incluyendo ropa para mujeres, hombres y niños. (Topitop, s.f.)

El crecimiento sostenido de Topitop ha demandado la conciliación de información contable y financiera cada vez más eficientes y robustos, alineados con los estándares internacionales del sector retail. Como señala López (2021), “las empresas del sector retail requieren sistemas de información contable integrados que permitan una gestión eficiente de inventarios, ventas y finanzas, así como una adecuada toma de decisiones” (López, 2021). En el caso de Topitop, esto se refleja en la necesidad de integrar sus sistemas internos y externos para lograr una consolidación oportuna y precisa de la información financiera.

1.2. Organigrama de la empresa

Figura 1

Organigrama de la empresa Topitop.



Nota. Elaboración propia.

1.3. Planteamiento del problema

1.3.1. Descripción del problema

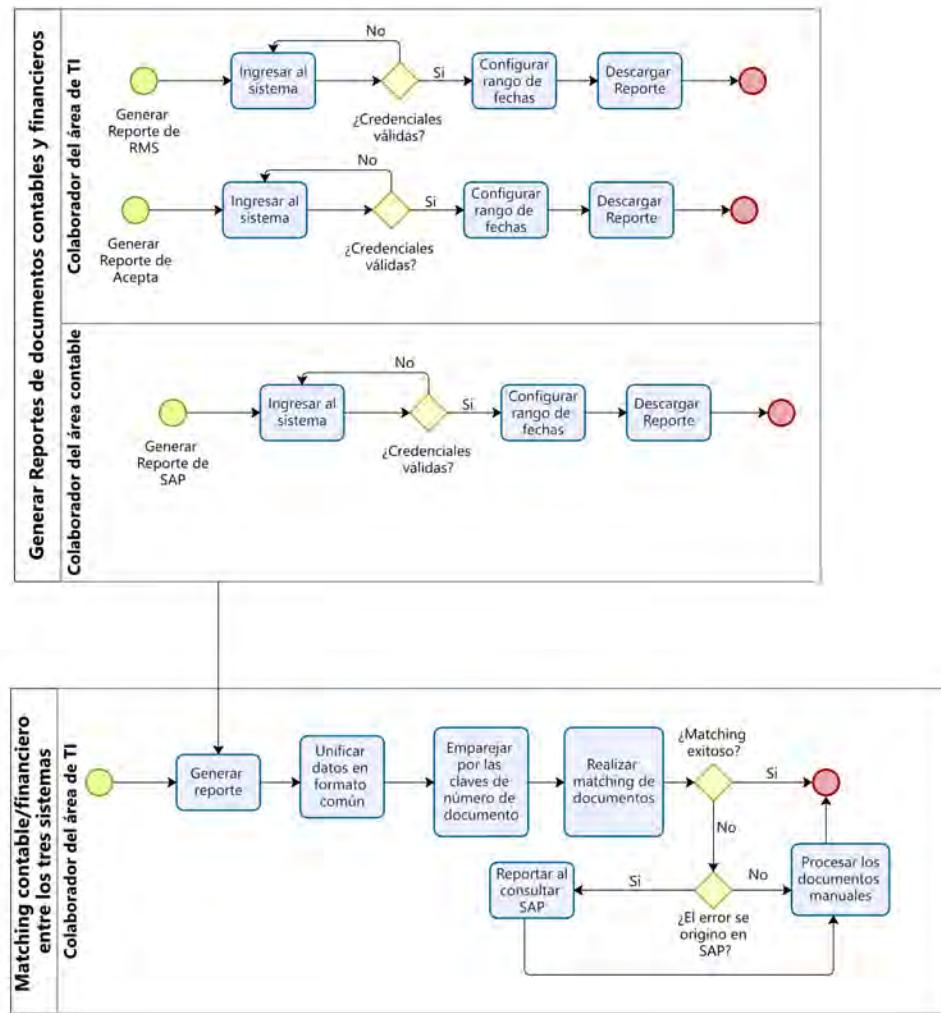
La empresa Topitop enfrenta demoras en la revisión, consolidación y sincronización de sus documentos contables y financieros debido a que el proceso se realiza de forma manual, lo que incrementa la probabilidad de errores. Esta gestión involucra tanto a los sistemas internos, como el Retail Management System (RMS) y el sistema de planificación de recursos empresariales (ERP) SAP, utilizados para las operaciones de back office, y el portal web externo Acepta, que actúa como intermediario en el envío de la información contable hacia la Superintendencia Nacional de Administración Tributaria (SUNAT). Esto genera problemas en diferentes ámbitos:

En el ámbito tributario, la omisión de la declaración de documentos contables tales como: facturas, boletas, notas de crédito y notas de débito. En los plazos establecidos puede conllevar al pago de multas y sanciones, según lo establecido por la Superintendencia Nacional de Administración Tributaria y el Código Tributario peruano (SUNAT, 2023).

En el ámbito económico y administrativo, genera un aumento considerable en los costos operativos y una reducción de la eficiencia. Esto se debe a que el trabajo manual incrementa el tiempo necesario para consolidar la información, eleva la probabilidad de errores y retrasa la toma de decisiones basada en datos contables confiables. Como consecuencia, las empresas enfrentan dificultades para competir en el mercado, disminuyen su rentabilidad y destinan mayores recursos financieros al control y corrección de errores en vez de enfocarlos en actividades productivas o estratégicas (Hurtado-Guevara, 2024).

Los siguientes diagramas muestran un resumen de la situación actual de los procesos manuales:

Figura 2
Proceso manual de los colaboradores



Nota. Elaboración propia.

Tabla 1
Detalle del tiempo invertido en los procedimientos manuales

Procedimiento	Tiempo (minutos)	Período (días)
Generar reporte de RMS	40	1
Generar reporte de ACEPTA	40	1
Generar reporte de SAP	40	1
Realizar comparación de los documentos	60	1
Total	180	—

Nota. Elaboración propia.

El diagrama describe el proceso manual de generación de reportes y conciliación contable entre tres sistemas principales de la empresa (RMS, Acepta y SAP), involucrando tanto al área de TI como al área contable/financiera. Permitiendo la detección, gestión de errores y sincronización de los documentos contables. Conjuntamente se detalla en un cuadro resumen el tiempo invertido de los procedimientos, lo cual demora en promedio 3 horas de trabajo diarias.

1.3.2. Identificación del problema

Topitop enfrenta un problema crítico en la gestión de su información contable y financiera: la revisión, consolidación y sincronización de los documentos contables se realiza de manera manual entre sus principales sistemas internos y externos (RMS, SAP y ACEP-TA). De acuerdo con López (2021), “la gestión manual en los procesos contables genera un incremento en los errores y dificulta la integración eficiente de la información financiera” (López, 2021, p. 50). Asimismo, Zhang et al. (2023) sostienen que “la ausencia de automatización en la consolidación de datos contables provoca retrasos y eleva los costos operativos en las organizaciones” (Zhang et al., 2023, p. 12). Estas limitaciones impactan negativamente en el cumplimiento tributario, la eficiencia administrativa y la competitividad de la empresa, por lo que resulta imprescindible “implementar soluciones tecnológicas que permitan optimizar la eficiencia operativa y reducir la incidencia de errores humanos” (Hurtado-Guevara, 2024, p. 28).

1.4. Formulación del problema

1.4.1. Problema General

¿Cómo lograr que la integración entre RMS, SAP y ACEPTA en Topitop permita una gestión contable y financiera eficiente, reduciendo errores de conciliación, retrasos en los

procesos y riesgos de incumplimiento tributario?

1.4.2. Problemas Específicos

- ¿Cómo eliminar los errores humanos en la conciliación de la información contable y financiera entre RMS, SAP y ACCEPTA asociados a procesos manuales?
- ¿Cómo reducir los retrasos operativos derivados de procesos manuales de revisión de documentos contables y financieros?
- ¿Cómo minimizar los riesgos de incumplimiento tributario derivados de inconsistencias en la información?

1.5. Objetivos

1.5.1. Objetivo General

Implementar un sistema que automatice la gestión de la información contable y financiera en la empresa Topitop, mediante la integración de tecnología RPA, procesos ETL y servicios de AWS, con el fin de mejorar la eficiencia operativa y mitigar errores asociados a tareas manuales.

1.5.2. Objetivos Específicos

- Eliminar los errores humanos de conciliación mediante la integración automatizada y validación de datos entre RMS, SAP y ACCEPTA.
- Disminuir los tiempos operativos automatizando tareas repetitivas del proceso de conciliación mediante RPA y procesamiento ETL en AWS.

- Minimizar el riesgo tributario mediante la estandarización y envío de alertas automáticas sobre inconsistencias, garantizando oportunidad del envío de información.
- Evaluar el impacto del sistema en términos de eficiencia operativa y reducción de errores.

1.6. Justificación

- Mejorar el control de documentos.
- Mejorar y/u operativizar de forma eficiente el control financiero.
- Brindar una solución eficiente y mejorar los procesos de la organización.
- Reducir procesos manuales de conciliación de documentos entre los usuarios en la empresa.
- Minimizar los errores de los usuarios al declarar documentos contables a la SUNAT.
- El software permitirá consolidar documentos de manera más rápida y precisa, lo que mejorará su eficiencia y reducirá los errores.

1.7. Presupuesto

Tabla 2

Presupuesto para el desarrollo del proyecto

Denominación	Costo (S/.)
Compra de laptop	3800.00
Servicio de energía eléctrica	800.00
Servicio de internet	640.00
Infraestructura AWS (8 meses)	10,000.00
Imprevistos (10 %)	1524.00
Total	16764.00

Nota. Montos expresados en soles (S/.). Elaboración propia.

1.8. Contribuciones

El presente proyecto, contribuye con la automatización eficiente de tareas contables y financieras, integración de datos para una visión completa, mayor precisión en informes, toma de decisiones informadas, reducción de costos operativos, cumplimiento normativo, mejora de la competitividad y escalabilidad. Todo lo cual fortalecerá la gestión contable y financiera, impulsará la eficiencia y la competitividad, y facilitará el crecimiento continuo de la empresa.

1.9. Impacto Social

- Mejora en la calidad de trabajo: La automatización de tareas repetitivas puede liberar a los empleados de cargas de trabajo monótonas, permitiéndoles centrarse en tareas más estratégicas y creativas, lo que puede aumentar la satisfacción laboral

y la calidad de vida en el trabajo.

- Acceso a datos transparentes: La disponibilidad de datos precisos y actualizados puede mejorar la transparencia en la gestión contable y financiera, lo que puede tener un impacto positivo en la confianza de los inversores y socios comerciales.
- Mejora en la eficiencia empresarial: Una empresa más eficiente y competitiva puede estar en una mejor posición para crecer y expandirse, lo que puede tener un impacto positivo en la economía local y en la comunidad en general.
- Cumplimiento normativo: La implementación de un sistema contable y financiero más robusto puede ayudar a la empresa a cumplir mejor con las regulaciones financieras y fiscales, lo que puede tener un impacto positivo en la integridad del mercado y la recaudación de impuestos.
- Desarrollo de habilidades: La inversión en tecnología avanzada puede fomentar el desarrollo de habilidades técnicas entre los empleados, lo que puede ser beneficioso para su empleabilidad futura.

Capítulo 2

Metodología

2.1. Alcance

La presente investigación es de tipo aplicada, ya que su finalidad principal es resolver un problema concreto en la gestión contable y financiera de la empresa Topitop, mediante el desarrollo e implementación de un sistema informático. Según Hernandez et al. (2014), la investigación aplicada se orienta a la utilización de conocimientos teóricos y técnicos existentes para intervenir directamente en una realidad específica, generando soluciones prácticas evaluables en el contexto real de la organización.

En este tipo de investigación no resulta indispensable la formulación de hipótesis, pues el objetivo central no es comprobar relaciones causales, sino diseñar, implementar y validar una solución tecnológica que permita optimizar los procesos actuales (Bisquerra, 2014). La validación se realiza mediante la descripción y el análisis comparativo de la situación antes y después de la implementación del sistema, lo cual permite identificar los cambios y beneficios obtenidos sin requerir comprobación estadística de hipótesis (Hernandez et al., 2014).

2.2. Diseño metodológico

El diseño metodológico integra tres dimensiones:

2.2.1. Diseño de la investigación

El diseño es no experimental y longitudinal, dado que no se manipulan variables independientes, sino que se observa el comportamiento de los procesos en su contexto real, evaluando los cambios antes y después de la implementación (Hernandez et al., 2014). Este tipo de diseño es adecuado cuando el objetivo es analizar la evolución de fenómenos en el tiempo sin alterar las condiciones en las que ocurren.

2.2.2. Tipo de la investigación

La investigación es de tipo aplicada y descriptiva-comparativa. Es aplicada porque se orienta a resolver un problema práctico en la organización (Cook & Campbell, 2005), y es descriptiva-comparativa porque se analizan las condiciones de los procesos contables y financieros en dos escenarios: el previo y el posterior a la implementación del sistema (Tamayo, 2011).

2.2.3. Método de investigación

Se emplea un método comparativo, organizado en tres fases principales:

1. Diagnóstico: análisis de la situación actual de conciliación contable y financiera entre RMS, SAP y ACEPTA.
2. Implementación: desarrollo del sistema con RPA, ETL y servicios de AWS, siguiendo la metodología Scrum.

3. Validación: evaluación de resultados a partir de indicadores de eficiencia, reducción de errores y tiempos de procesamiento.

El método comparativo permite establecer diferencias y semejanzas entre los estados inicial y final de los procesos, constituyendo un recurso válido para evaluar cambios en contextos organizacionales (Ardila, 2011).

2.3. Para el desarrollo del sistema

El sistema de gestión e integración de información contable y financiera basado en RPA, ETL y AWS para la empresa Topitop es una solución tecnológica que automatiza la gestión de información contable y financiera, aumenta la precisión y mejora la toma de decisiones al proporcionar información financiera confiable y actualizada. Este sistema se sustenta en el uso de bots de software, el análisis de grandes volúmenes de datos y una infraestructura en la nube, centralizando y consolidando la información en un único sistema (Aggarwal, 2015; Amazon Web Services, 2020).

En cuanto a la metodología de desarrollo, se adoptó el marco de trabajo ágil Scrum. Scrum es un proceso iterativo e incremental en el que se aplican buenas prácticas de trabajo colaborativo para obtener los mejores resultados posibles en proyectos complejos (Schwaber & Sutherland, 2020).

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto, lo cual permite obtener resultados tempranos y medibles en contextos de alta incertidumbre (Pressman, 2014). Por ello, Scrum es especialmente adecuado para proyectos donde los requisitos son cambiantes o poco definidos, y donde la flexibilidad y la productividad son fundamentales.

Debido a estos factores se eligió Scrum como marco de trabajo, ya que permite obtener un producto mínimo viable en el menor tiempo posible sin alterar el flujo de negocio. Esta

metodología ofrece alta adaptabilidad, fomenta la cooperación de las áreas involucradas (finanzas, contabilidad y TI) y facilita la entrega continua de valor. De esta forma, se asegura que la documentación y la información de los procesos estén disponibles de manera oportuna y confiable, aumentando la probabilidad de éxito del proyecto (Sommerville, 2011).

Capítulo 3

Marco Teórico

3.1. Antecedentes

3.1.1. Antecedentes Internacionales

Zhang et al. (2023). *Robotic Process Automation (RPA) Implementation Case Studies in Accounting: A Beginning to End Perspective*. Accounting Horizons. [Traducción propia].

Problema: Las empresas enfrentan la problemática de depender de la subcontratación de procesos contables y de la ejecución manual de tareas rutinarias y repetitivas, lo que genera falta de control, ineficiencias operativas, aumento de errores y limita la capacidad del personal contable y de los auditores para dedicarse a actividades estratégicas y de mayor valor para la organización.

Objetivo: Investigar cómo la implementación de RPA en procesos contables puede mejorar la eficiencia, el control y la calidad de la información, permitiendo a las organizaciones reducir la dependencia de la subcontratación y liberar a su personal para actividades de mayor valor.

Conclusión: La implementación de RPA demostró ser exitosa principalmente al mejorar la eficiencia operativa, el control de los procesos y la satisfacción del personal y liberando a los empleados de labores repetitivas. Sin embargo, el éxito de la automatización depende de una adecuada gestión del cambio, comunicación efectiva, mantenimiento continuo de los bots y una evaluación integral que considere tanto los beneficios cuantitativos como cualitativos, más allá de la simple reducción de costos.

Axmann & Harmoko (2022). *Process & Software Selection for Robotic Process Automation (RPA)*. Technical Journal / Tehnicki Glasnik. [Traducción propia].

Problema: Muchas empresas, pese a la presión por ser más eficientes y rentables, carecen de un método claro para identificar qué procesos automatizar con RPA y qué software elegir, lo que provoca que la automatización no siempre se enfoque en las actividades de mayor valor, limitando así los beneficios de la inversión tecnológica.

Objetivo: Desarrollar y aplicar un método sistemático y transparente para identificar y seleccionar los procesos internos más adecuados para ser automatizados y el software adecuado de RPA, con el fin de maximizar la rentabilidad y eficiencia operativa de la empresa.

Conclusiones: Para seleccionar y automatizar procesos con RPA de manera efectiva, primero se realiza una observación directa y sesiones de brainstorming con los responsables de cada área para identificar procesos repetitivos, manuales y que consumen muchos recursos. Estos procesos se evalúan usando criterios clave como frecuencia de ejecución, tiempo requerido, estabilidad y grado de automatización actual, asignando una puntuación a cada uno. Luego, se suman las puntuaciones para comparar objetivamente y seleccionar el proceso que mayor beneficio aportaría al automatizarlo. Posteriormente, se definen criterios para elegir el software de RPA más adecuado, considerando aspectos como facilidad de uso, integración, costos, seguridad y escalabilidad. Se ponderan estos criterios mediante el método de comparación por pares y se comparan las opciones de software utilizando

un análisis de valor-beneficio. Finalmente, se mide el impacto de la automatización en términos de ahorro de tiempo, reducción de costos y mejora en la precisión.

Machado et al. (2019). *DOD-ETL: distributed on-demand ETL for near real-time business intelligence*. [Traducción propia].

Problema: El problema es la lentitud o falta de inmediatez del proceso ETL tradicional para proveer información útil en tiempo real o casi en tiempo real.

Objetivo: Proponer y evaluar una nueva herramienta llamada que permita superar las limitaciones de los procesos ETL tradicionales, brindando una solución capaz de realizar la integración, transformación y carga de datos en tiempo casi real, con alta disponibilidad, baja latencia y escalabilidad horizontal para aplicaciones de Business Intelligence (BI).

Conclusion: La nueva herramienta DOD-ETL (Distributed On-Demand ETL) logró procesar cargas de trabajo hasta 10 veces más rápido que frameworks de procesamiento de flujo tradicionales, como Spark Streaming sin personalizaciones. Por ejemplo, pudo procesar más de 10,000 registros por segundo en pruebas sintéticas, frente a solo 1,230 registros por segundo del Spark Streaming convencional. Logra esto combinando una canalización de flujo de datos bajo demanda con una arquitectura distribuida, paralela e independiente de la tecnología, que utiliza almacenamiento en memoria (in-memory caching) y una partición eficiente de los datos.

3.1.2. Antecedentes Nacionales

Salas (2023). *Sistema que automatice el proceso de cuadre diario de las transferencias interbancarias inmediatas de los bancos en el Perú*. Universidad Peruana de Ciencias Aplicadas.

Problema: En el contexto de las Transferencias Interbancarias Inmediatas (TIN) en el

Perú, los bancos enfrentan la necesidad de garantizar la precisión y la integridad de sus transacciones financieras mediante un proceso diario de cuadre o conciliación de datos entre sus propias bases de datos y los registros proporcionados por la Cámara de Compensaciones Electrónicas (CCE). Sin embargo, este proceso de comparación y conciliación se realiza tradicionalmente de manera manual, lo que lo hace propenso a errores humanos, demoras en la detección de discrepancias y respuestas ineficientes a las partes interesadas. La gestión manual no solo incrementa el riesgo de inconsistencias en la información financiera, sino que también afecta negativamente la transparencia, la confianza y la eficiencia operativa del sistema bancario.

Objetivo: Implementar un proceso Batch que permita consultar y extraer automáticamente la información relevante tanto de las bases de datos bancarias como de los registros de la Cámara de Compensaciones Electrónicas (CCE); aplicar procesos ETL (Extract, Transform, Load) a los datos provenientes de la CCE para asegurar la calidad y coherencia de la información a conciliar; contrastar y conciliar de manera automática los datos de ambas fuentes, identificando y reportando oportunamente las discrepancias encontradas; y, finalmente, reducir los tiempos de ejecución y la intervención manual en el proceso de cuadre, mejorando así la eficiencia operativa y la respuesta ante las partes interesadas.

Conclusión: Al implementar un sistema automatizado, el cual emplea diversas técnicas de tecnología de información (TI), representa un avance significativo en la gestión del cuadre de las TIN en el Perú. Este enfoque permite una conciliación más eficiente y precisa entre las diversas datas mencionadas, reduciendo el riesgo de errores humanos y garantizando la integridad de las operaciones financieras. Asimismo, esta mejora no solo agiliza los procedimientos, sino que también fortalece la confianza y transparencia en el sistema bancario, beneficiando tanto a las organizaciones como a los clientes.

Palacios Martel (2022). *Sistema ETL para la mejora en el movimiento de información entre servidores no comunicados en el Data Warehouse MIS de Financiera Confianza S.A.A.* Universidad Nacional del Centro del Perú.

Problema: La Financiera Confianza, en el marco del proyecto MIS, se requiere transferir información entre servidores satélites no productivos y nuevos servidores alojados en la nube Azure de Microsoft. Actualmente, el traspaso de información se realiza de manera manual, lo que genera una serie de ineficiencias operativas como tiempos muertos prolongados, demoras en la disponibilidad de los datos, alta dependencia del personal técnico y una mayor probabilidad de errores humanos.

Objetivo: Diseñar una solución automatizada que permita mejorar el proceso de movimiento de información entre servidores aislados, optimizando tiempos y reduciendo los errores derivados del manejo manual de datos.

Conclusión: La tesis implementó un sistema ETL automatizado que permitió mejorar en un 64.93 % el tiempo promedio de transferencia de información entre servidores no comunicados en Financiera Confianza, eliminando los tiempos muertos y reduciendo la intervención manual. Para lograrlo, se utilizaron tecnologías como SQL Server Integration Services (SSIS) para la ejecución de paquetes automatizados, Java para el desarrollo de aplicaciones que gestionan la recuperación y carga de archivos, servicios de Google (como Gmail o Google Drive) para la transferencia de archivos CSV, y Microsoft Azure como entorno de destino en la nube donde se centraliza la información. Esta solución demostró ser efectiva para optimizar la integración de datos en entornos con restricciones de conectividad directa.

Ochoa Surco & Osorio Schuler (2022). *Implementación de un Robotics Process Automation (RPA) para mejorar el proceso de validación de estados de cuenta en una entidad financiera*. Universidad Tecnológica del Perú.

Problema: El problema identificado en la investigación radica en que el proceso de validación de estados de cuenta en una entidad financiera se realizaba de forma manual, lo que implicaba una alta carga operativa, consumo excesivo de tiempo y riesgo de errores humanos. Un equipo de 10 personas debía revisar manualmente una muestra aleatoria

de los estados generados en cada ciclo de facturación, lo que limitaba la cobertura de validación y afectaba la eficiencia y calidad del proceso.

Objetivo: Mejorar el proceso de validación de los estados de cuenta en una entidad financiera mediante la implementación de Robotic Process Automation (RPA).

Conclusión: La implementación de RPA permitió mejorar los indicadores definidos, tales como: reducción del 35.7 % en el tiempo de validación de estados de cuenta, ahorro de 602.4 horas hombre mensuales, se consiguió eliminar el error operativo y se logró incrementar en 25 % el porcentaje de campos validados en el estado de cuenta, alcanzado el porcentaje ideal del 100 %.

Lizana (2018). *Automatización en la recolección, tratamiento y envío de información estadística médico-asistencial a la Superintendencia Nacional de Salud (SUSALUD) basado en procesos de ETL y RPA para la clínica Adventista Ana Stahl*, Universidad Peruana Unión.

Problema: El proceso manual de recolección, tratamiento y envío de información estadística médica-asistencial a SUSALUD en la Clínica Ana Stahl presentaba constantes errores en los archivos *.txt validados, requería un excesivo tiempo para generar reportes y demandaba una alta carga administrativa, debido a tareas repetitivas y poco eficientes, lo que afectaba la calidad, precisión y oportunidad en el cumplimiento de las obligaciones informativas con el ente regulador.

Objetivo: Implementar una solución de software basada en procesos ETL y RPA para automatizar el envío de información estadística médica-asistencial a SUSALUD en la Clínica Ana Stahl.

Conclusión: Se logró reducir significativamente el tiempo de procesamiento, disminuir los errores en los archivos .txt, y eliminar tareas repetitivas, mejorando la calidad de los

datos y la eficiencia operativa.

3.2. Bases Teóricas

3.2.1. Sistema de Gestión Contable

Los Sistemas de Contabilidad de Gestión han sido clasificados de distintas maneras, como sistemas contables de gestión, sistemas de información y control de gestión, o sistemas de contabilidad de gestión. A pesar de las variaciones en la terminología, todas estas designaciones se refieren esencialmente al mismo concepto, ya que comparten semejanzas en sus respectivas definiciones. Un sistema de información contable se define como un conjunto coordinado de elementos que abarcan hardware, software, bases de datos, procedimientos y modelos. Su objetivo principal consiste en analizar y procesar información administrativa para respaldar la toma de decisiones (Gelinas et al., 2018).

En cuanto al Sistema de Gestión Contable, comprende una serie de procedimientos y métodos destinados a suministrar a los miembros internos de las organizaciones la información necesaria en términos de cantidad y calidad para planificar acciones y operaciones que conduzcan al logro de los objetivos establecidos. La información generada es altamente dinámica y se ajusta a las necesidades específicas de cada integrante interno, dependiendo de su posición en la estructura organizativa. Los sistemas de información se han vuelto fundamentales en la administración de las empresas actuales, adquiriendo un rol estratégico en las organizaciones del siglo XXI, evolucionando de ser simplemente herramientas de automatización de procesos operativos a convertirse en elementos cruciales a considerar en la formulación de estrategias empresariales (Varela et al., 2018).

Es el conjunto de métodos, procedimientos y recursos que las entidades emplean para llevar a cabo el control y el registro de sus actividades comerciales y empresariales y resumirlas a través de informes útiles para el uso por los distintos agentes económicos

interesados (Abolacio Bosch, 2013).

3.2.2. Conciliación Bancaria

La conciliación bancaria es un procedimiento contable mediante el cual las empresas comparan y verifican sus registros internos de movimientos financieros con los extractos emitidos por las entidades bancarias, con el fin de identificar discrepancias, errores u operaciones pendientes como depósitos en tránsito o cheques no cobrados. Este proceso resulta fundamental para garantizar la exactitud de la información financiera, facilitar la detección de fraudes o irregularidades, cumplir con obligaciones fiscales y regulatorias, y proveer datos confiables para la toma de decisiones y auditorías internas o externas (BBVA Spark, 2023a).

3.2.3. Sistema de Gestión Financiera

Los sistemas de gestión financiera, diseñados para generar, conectar, registrar y transmitir diversas operaciones monetarias, tienen como objetivo central la gestión efectiva del capital. En resumen, sin algún tipo de software contable, una entidad no podría mantenerse en funcionamiento. Uno de los requisitos más básicos para cualquier organización es contar con sistemas que supervisen y regulen los flujos de dinero entrantes y salientes. Sin embargo, un sistema de administración financiera verdaderamente eficaz puede ir más allá: mejorar la rentabilidad, evaluar el flujo de fondos, calcular las obligaciones fiscales, garantizar el cumplimiento de normativas y asegurar la viabilidad a largo plazo de la empresa (Oracle, s.f.).

Un sistema para gestión financiera es el software y los procesos que se usan para manejar los ingresos, gastos y activos de una organización. Además de dar soporte a las operaciones financieras diarias, el objetivo de un sistema para gestión financiera es maximizar las ganancias y garantizar la sostenibilidad empresarial a largo plazo.

Ayuda a los equipos financieros a:

- Optimizar la facturación y el cobro de facturas.
- Optimizar el flujo de efectivo diario, mensual y anual.
- Mantener seguimientos de auditoría y cumplir con las regulaciones contables.
- Automatizar los procesos financieros y reducir los errores de contables.
- Brindar mejores presupuestos, proyecciones y planificación.
- Acelerar el cierre financiero y las actividades de informes (SAP, s.f.).

El software para gestión financiera puede ser parte de un sistema de planificación de recursos empresariales (ERP), el cual consolida los datos financieros y operativos y brinda a los equipos una visión integral del negocio. Las aplicaciones financieras independientes también se pueden combinar para crear un sistema de gestión financiera de punta a punta (Oracle, s.f.).

3.2.4. Gestión de tesorería

La gestión de tesorería es el proceso estratégico mediante el cual una empresa supervisa, planifica y controla sus flujos de efectivo —entradas y salidas— para asegurar la disponibilidad de recursos suficientes que permitan cumplir con sus obligaciones financieras, optimizar la liquidez e invertir los excedentes de manera rentable, al tiempo que se minimizan los riesgos financieros y los costos asociados (BBVA Spark, 2023b).

3.2.5. Cumplimiento fiscal y regulatorio

El cumplimiento fiscal y regulatorio es un proceso esencial mediante el cual una empresa garantiza que sus operaciones y registros contables se ajustan a la normativa tributaria,

legal y ética vigente, proporcionando transparencia y permitiendo evitar sanciones económicas, daños reputacionales y conflictos con las autoridades; además, un enfoque integral de cumplimiento fortalece la planificación financiera y facilita el acceso a beneficios regulatorios o fiscales (EY, 2025).

3.2.6. ACEPTA

ACEPTA es una plataforma de servicios electrónicos que actúa como un *Proveedor de Servicios Electrónicos (PSE)* autorizado por la Superintendencia Nacional de Administración Tributaria (SUNAT) en Perú. Su función principal es facilitar el envío, validación, firma digital y almacenamiento de comprobantes electrónicos, como facturas, boletas, notas de crédito y débito, asegurando que las empresas cumplan con las normativas tributarias vigentes. La empresa opera bajo los lineamientos técnicos y legales establecidos por la SUNAT, y su infraestructura está diseñada para integrarse con sistemas empresariales mediante APIs, garantizando alta disponibilidad, seguridad y trazabilidad en el procesamiento de documentos electrónicos (ACEPTA Perú, s.f.).

Dentro de las organizaciones, ACEPTA se posiciona como un puente entre los sistemas de facturación internos y la SUNAT. Esta integración permite automatizar el envío de comprobantes electrónicos, lo que reduce la intervención humana, disminuye el margen de error y optimiza los tiempos de respuesta frente a las obligaciones fiscales. En empresas con gran volumen de operaciones diarias, como Topitop, el uso de ACEPTA es crucial para mantener la eficiencia operativa, asegurar el cumplimiento tributario y simplificar la gestión documental (ACEPTA Perú, s.f.).

Ventajas de utilizar ACEPTA:

- Automatización del proceso de emisión y envío de comprobantes electrónicos.
- Cumplimiento con las normativas establecidas por la SUNAT.
- Seguridad, trazabilidad y respaldo digital mediante firma electrónica.

- Integración con sistemas contables mediante servicios web (APIs).
- Generación de reportes en tiempo real y soporte técnico especializado (ACEPTA Perú, s.f.).

3.2.7. Scrum

Según Schwaber & Sutherland (2020), Scrum es un marco de trabajo por el cual las personas pueden abordar problemas complejos adaptativos, a la vez que entregar productos del máximo valor posible productiva y creativamente.

Scrum es:

- Liviano.
- Fácil de entender.
- Difícil de llegar a dominar.

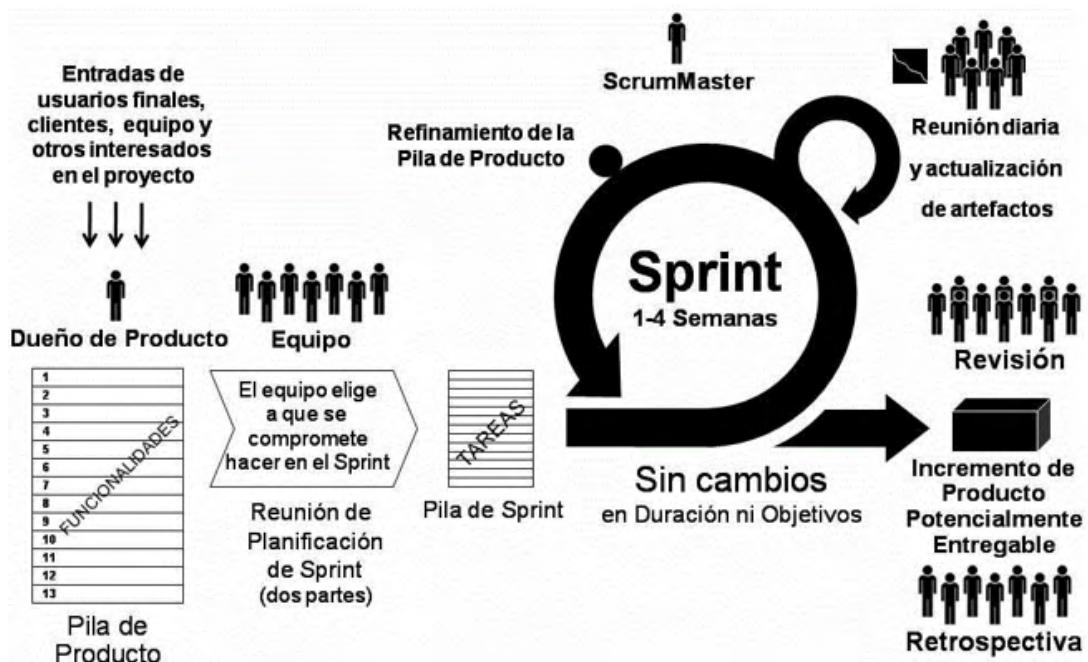
Scrum es un marco de trabajo de procesos que ha sido usado para gestionar el desarrollo de productos complejos desde principios de los años 90. Scrum no es un proceso o una técnica para construir productos; en lugar de eso, es un marco de trabajo dentro del cual se pueden emplear varios procesos y técnicas. Scrum muestra la eficacia relativa de las prácticas de gestión de producto y las prácticas de desarrollo de modo que podamos mejorar. El marco de trabajo Scrum consiste en los Equipo Scrum y sus roles, eventos, artefactos y reglas asociadas. Cada componente dentro del marco de trabajo sirve a un propósito específico y es esencial para el éxito de Scrum y para su uso. Las reglas de Scrum relacionan los eventos, roles y artefactos, gobernando las relaciones e interacciones entre ellos (Schwaber & Sutherland, 2020).

Scrum se caracteriza por ser un método progresivo y sistemático para el desarrollo de proyectos, productos y aplicaciones, destacando por su enfoque iterativo e incremental. Este modelo organiza las tareas en ciclos llamados "Sprints", que tienen una duración fija

de 1 a 4 semanas, sucediéndose en secuencia sin prolongarse más allá de la fecha límite. Al inicio de cada Sprint, un equipo flexible elige elementos prioritarios de una lista y se compromete a finalizarlos al término del ciclo. Durante el Sprint, no se admiten modificaciones en los elementos seleccionados. A diario, el equipo tiene una breve reunión para compartir progresos y actualizar gráficos sencillos que sirven como referencia para evaluar el trabajo pendiente. Al concluir el Sprint, el equipo repasa sus logros con los involucrados en el proyecto, mostrando lo que han construido y recopilando valiosos comentarios para integrar en el próximo ciclo. Scrum pone gran énfasis en entregar productos completamente funcionales al término de cada Sprint, lo que implica que el código esté integrado y sometido a rigurosas pruebas (Schwaber & Sutherland, 2020).

Un tema importante en Scrum es “inspeccionar y adaptar”. El desarrollo inevitablemente implica aprender, innovación y sorpresas. Por eso Scrum hace hincapié en dar un pequeño paso de desarrollo; inspeccionar el producto resultante y la eficacia de las prácticas actuales; y entonces adaptar el objetivo del producto y las prácticas del proceso. Y volver a repetir (Deemer et al., 2009).

Figura 3
Scrum framework



Nota. Información básica de Scrum (*The Scrum Primer*). Deemer, Benefield, & Vodde (2009)..

Scrum está conformado por Equipo Scrum, Eventos de Scrum y Artefactos de Scrum.

3.2.7.1. Equipo Scrum

El Equipo Scrum consiste en un Dueño de Producto (Product Owner), el Equipo de Desarrollo (Development Team) y un Scrum Master. Los Equipos Scrum son auto-organizados y multifuncionales. Los equipos autoorganizados eligen la mejor forma de llevar a cabo su trabajo y no son dirigidos por personas externas al equipo. Los equipos multifuncionales tienen todas las competencias necesarias para llevar a cabo el trabajo sin depender de otras personas que no son parte del equipo. El modelo de equipo en Scrum está diseñado para optimizar la flexibilidad, la creatividad y la productividad. Los Equipos Scrum entregan productos de forma iterativa e incremental, maximizando las oportunidades de obtener retroalimentación. Las entregas incrementales de producto “Terminado” aseguran que siempre estará disponible una versión potencialmente útil y funcional del producto (Schwaber & Sutherland, 2020).

El Dueño de Producto (Product Owner)

El Dueño del Producto es el responsable de maximizar el valor del producto y el trabajo del Equipo de Desarrollo. El Dueño de Producto es la única persona responsable de gestionar la Lista del Producto (Product Backlog), que incluye expresar claramente los elementos, ordenarlos, optimizar el valor, garantizar visibilidad y transparencia, y asegurar la comprensión por parte del equipo. Aunque estas tareas pueden delegarse, la responsabilidad final siempre recae en el Dueño de Producto. Es indispensable que la organización respete sus decisiones (Schwaber & Sutherland, 2020).

El Equipo de Desarrollo (Development Team)

El Equipo de Desarrollo está compuesto por profesionales encargados de entregar un Incremento de producto “Terminado” al final de cada Sprint. Son autoorganizados, multifuncionales y no reconocen títulos ni subequipos. La responsabilidad del trabajo recae en el equipo como un todo. El tamaño ideal del equipo es de entre 3 y 9 personas (Schwaber & Sutherland, 2020).

El Scrum Master

El Scrum Master es el responsable de asegurar que Scrum sea entendido y adoptado. Es un líder al servicio del equipo, facilita interacciones efectivas con externos y enseña cómo aplicar las reglas y prácticas de Scrum correctamente (Schwaber & Sutherland, 2020).

3.2.7.2. Eventos de Scrum

Scrum define eventos como el Sprint, la Planificación del Sprint, el Scrum Diario, la Revisión del Sprint y la Retrospectiva del Sprint. Estos eventos son bloques de tiempo que fomentan la inspección y adaptación, pilares fundamentales de Scrum (Schwaber & Sutherland, 2020).

El Sprint

Es un bloque de tiempo de hasta un mes en el que se desarrolla un incremento de producto. Cada Sprint incluye todos los eventos de Scrum y permite la predictibilidad del progreso y la reducción de riesgos (Schwaber & Sutherland, 2020).

Planificación de Sprint

El trabajo a realizar durante el Sprint se planifica en la Planificación de Sprint. Este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo. La Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes. Para Sprints más cortos el evento es usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo (Schwaber & Sutherland, 2020).

Schwaber & Sutherland (2020), afirma que la planificación de Sprint responde a las siguientes preguntas:

- ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?

Objetivo del Sprint (Sprint Goal)

El Objetivo del Sprint es una meta establecida para el Sprint que puede lograrse mediante la implementación de la Lista de Producto. Proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento. Se crea durante la Planificación del Sprint. El objetivo del Sprint brinda al equipo de desarrollo cierta flexibilidad con respecto a la funcionalidad implementada en el Sprint. Los elementos de la Lista del Producto seleccionados ofrecen una función coherente que puede ser el objetivo del Sprint. El objetivo del Sprint puede representar otro nexo de unión que haga que el Equipo de Desarrollo trabaje en conjunto y no en iniciativas separadas. A medida que el equipo de desarrollo trabaja mantiene el objetivo del Sprint en mente. Con el fin de satisfacer el objetivo del Sprint se implementa la funcionalidad y la tecnología. Si el trabajo resulta ser diferente de lo que el Equipo de Desarrollo espera, ellos colaboran con el Dueño del Producto para negociar el alcance de la Lista de pendientes del Sprint (Sprint Bac-

klog)(Schwaber & Sutherland, 2020).

Scrum Diario (Daily Scrum)

El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para que el Equipo de Desarrollo sincronice sus actividades y cree un plan para las siguientes 24 horas. Esto se lleva a cabo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección acerca del trabajo que podría completarse antes del siguiente. El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad.(Schwaber & Sutherland, 2020).

Durante la reunión, cada miembro del Equipo de Desarrollo explica:

- ¿Qué hice ayer que ayudó al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Qué haré hoy para ayudar al Equipo de Desarrollo a lograr el Objetivo del Sprint?
- ¿Veo algún impedimento que evite que el Equipo de Desarrollo o yo logremos el Objetivo del Sprint? (Schwaber & Sutherland, 2020).

El Scrum Master se asegura de que el Equipo de Desarrollo tenga la reunión pero es el Equipo de Desarrollo el responsable de dirigir el Scrum Diario. El Scrum Master enseña al Equipo de Desarrollo a mantener el Scrum Diario en los límites del bloque de tiempo de 15 minutos. El Scrum Master se asegura de que se cumpla la regla de que solo los miembros del Equipo de Desarrollo participan en el Scrum Diario. Los Scrum Diarios mejoran la comunicación, eliminan la necesidad de realizar otras reuniones, identifican impedimentos a remover relativos al desarrollo, resaltan y promueven la toma de decisiones rápida y mejoran el nivel de conocimiento (Schwaber & Sutherland, 2020).

Revisión del Sprint

Al final del Sprint se lleva a cabo una Revisión de Sprint para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración. Se trata de una reunión restringida a un bloque de tiempo de cuatro horas para Sprints de un mes. Para Sprints más cortos, se reserva un tiempo usualmente más corto (Schwaber & Sutherland, 2020).

La Revisión de Sprint incluye los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados claves invitados por el Dueño de Producto.
- El Dueño de Producto explica qué elementos de la Lista de Producto se han “Terminado” y cuales no se han “Terminado”.
- El Equipo de Desarrollo habla acerca de qué estuvo bien durante el Sprint, qué problemas aparecieron y cómo fueron resueltos esos problemas.
- El Equipo de Desarrollo hace una demostración del trabajo que ha “Terminado” y responde preguntas acerca del Incremento.
- El Dueño de Producto habla acerca de la Lista de Producto en su estado actual. Proyecta fechas de finalización probables en el tiempo basándose en el progreso obtenido hasta la fecha (si es necesario).
- El grupo completo colabora acerca de qué hacer a continuación, de modo que la Revisión del Sprint proporcione información de entrada valiosa para Reuniones de

Planificación de Sprints subsiguientes.

- Revisión de la línea de tiempo, presupuesto, capacidades potenciales y mercado para la próxima entrega prevista del producto (Schwaber & Sutherland, 2020).

El resultado de la Revisión de Sprint es una versión actualizada de la Lista de Producto, que especifica los posibles elementos para el próximo Sprint. Es factible que la Lista de Producto también experimente ajustes generales para centrarse en nuevas posibilidades (Schwaber & Sutherland, 2020).

Retrospectiva del Sprint

La Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y de crear un plan de mejoras que sean abordadas durante el siguiente Sprint. La Retrospectiva de Sprint tiene lugar después de la Revisión de Sprint y antes de la siguiente Planificación de Sprint. Se trata de una reunión restringida a un bloque de tiempo de tres horas para Sprints de un mes. Para Sprints más cortos se reserva un tiempo usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. El Scrum Master participa en la reunión como un miembro del equipo ya que la responsabilidad del proceso Scrum recae sobre él (Schwaber & Sutherland, 2020).

El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas.
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras.
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo (Schwaber & Sutherland, 2020).

El Scrum Master alienta al equipo para que mejore, dentro del marco de proceso Scrum, su proceso de desarrollo y sus prácticas para hacerlos más efectivos y amenos para el siguiente Sprint. Durante cada Retrospectiva de Sprint, el Equipo Scrum planifica formas de aumentar la calidad del producto mediante la adaptación de la Definición de “Terminado” (Definition of “Done”) según sea conveniente. Para el final de la Retrospectiva de Sprint el Equipo Scrum debería haber identificado mejoras que implementará en el próximo Sprint. El hecho de implementar estas mejoras en el siguiente Sprint constituye la adaptación subsecuente a la inspección del Equipo de Desarrollo mismo. Aunque las mejoras pueden implementarse en cualquier momento, la Retrospectiva de Sprint ofrece un evento dedicado para este fin, enfocado en la inspección y la adaptación (Schwaber & Sutherland, 2020).

3.2.7.3. Artefactos de Scrum

Lista de Producto (Product Backlog)

Es la fuente única de requisitos. Es dinámica, ordenada y continuamente refinada por el Dueño de Producto en colaboración con el equipo. Los elementos deben estar preparados para ser “Terminados” en un Sprint (Schwaber & Sutherland, 2020).

Lista de Pendientes del Sprint (Sprint Backlog)

La Lista de Pendientes del Sprint es el conjunto de elementos de la Lista de Producto seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”. La Lista de Pendientes del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint (Schwaber & Sutherland, 2020).

Incremento

El Incremento es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la Definición de “Terminado” del Equipo Scrum. El incremento debe estar en condiciones de utilizarse sin importar si el Dueño de Producto decide liberarlo o no (Schwaber & Sutherland, 2020).

3.2.8. Automatización Robótica de Procesos (RPA)

La Automatización Robótica de Procesos (RPA, por sus siglas en inglés) es una tecnología que permite automatizar tareas digitales repetitivas mediante bots de software que imitan acciones humanas. Estos bots interactúan con interfaces digitales como lo haría un usuario humano, ejecutando procesos con rapidez y precisión (Automation Anywhere, 2025b).

Según IBM, la RPA utiliza software para automatizar tareas de back-office que son repetitivas, como ingresar datos, procesar transacciones o responder correos electrónicos, mejorando la productividad y reduciendo errores (IBM, 2025).

La Automatización Robótica de Procesos (RPA) se define de la siguiente manera:

Las herramientas de RPA llevan a cabo operaciones basadas en declaraciones [if, then, else] en estructuras de datos, generalmente mediante una combinación de interacciones en la interfaz de usuario o conectándose a una API para controlar servidores de clientes, mainframes o código HTML. Una herramienta de RPA opera mapeando un proceso en el lenguaje específico de la herramienta de RPA, permitiendo que un robot de software lo siga, y asignando un tiempo de ejecución para ejecutar el script a través de un tablero de control. Por lo tanto, el objetivo de las herramientas de RPA es aliviar la carga de tareas repetitivas y sencillas asignadas a los empleados (Van Der Aalst et al., 2018).

3.2.8.1. Componentes clave de RPA:

- **Bots de software:** Programas informáticos diseñados para realizar tareas específicas, como extraer datos o interactuar con sistemas empresariales.
- **Plataformas RPA:** Herramientas como UiPath, Automation Anywhere o Blue Prism que permiten desarrollar, implementar y monitorear bots.
- **Automatización basada en reglas:** Los bots ejecutan tareas siguiendo instrucciones lógicas (if/then/else), ideales para procesos estructurados y repetitivos (Wewerka & Reichert, 2020).

3.2.8.2. Principios clave:

- **No requiere modificar sistemas existentes:** La RPA opera sobre interfaces de usuario sin alterar la infraestructura actual (Automation Anywhere, 2025b).
- **Alta precisión y consistencia:** Los bots ejecutan tareas de forma uniforme, reduciendo significativamente errores humanos (IBM, 2025).
- **Productividad 24/7:** Pueden ejecutarse sin interrupción, aumentando la eficiencia y disponibilidad de los procesos.
- **Escalabilidad:** Es posible agregar más bots para soportar mayores volúmenes de trabajo según la demanda (Automation Anywhere, 2025a).

3.2.8.3. Tipos de RPA:

- **RPA atendida:** Asiste al usuario humano y se activa en conjunto con sus actividades.
- **RPA desatendida:** Funciona de forma autónoma sin interacción humana (Automation Anywhere, 2025a).

3.2.8.4. Aplicaciones de RPA en el Contexto Peruano

En el Perú, la adopción de la Automatización Robótica de Procesos (RPA) ha experimentado un crecimiento significativo, especialmente en sectores como banca, telecomunicaciones, retail y el ámbito público. Empresas como Interbank, BCP y Claro Perú han implementado soluciones automatizadas en áreas de atención al cliente, conciliaciones contables y gestión documental. Por ejemplo, Interbank ha desarrollado iniciativas de automatización en procesos financieros clave, mientras que Claro Perú ha optimizado su atención al cliente mediante la implementación de bots de software que replican tareas repetitivas.

Según el estudio "Nuevos horizontes de la madurez digital en el Perú" realizado por EY Perú en 2024, el 50 % de las empresas en el país utilizan regularmente herramientas tecnológicas en sus decisiones financieras, destacando la automatización de procesos contables como una de las tecnologías más empleadas. Este estudio también señala que el 64 % de las empresas han implementado soluciones de automatización en sus procesos contables, evidenciando una tendencia creciente hacia la digitalización y eficiencia operativa en el entorno empresarial peruano (EY Perú, 2024).

En el sector público, la Superintendencia Nacional de Administración Tributaria (SUNAT) ha adoptado procesos automatizados para mejorar la atención al contribuyente y la validación de comprobantes electrónicos. Estas iniciativas han permitido una mayor eficiencia en la gestión tributaria y una reducción en los tiempos de respuesta a los ciudadanos (SUNAT, 2021).

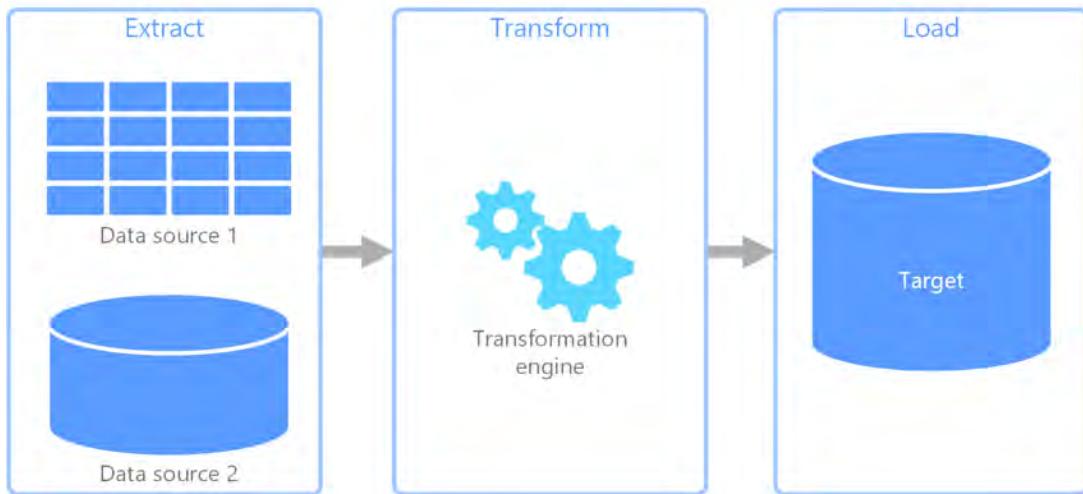
3.2.9. Extracción, transformación y carga (ETL)

ETL es el acrónimo de Extraer, Transformar y Cargar. ETL sirve como el motor impulsor para el análisis. Sin ETL, estaría confinado a la estructura original de los datos con los que comienza. En muchas situaciones, esto puede ser adecuado, pero a medida

que progrese en su análisis, querrá fusionar diversas fuentes de datos, combinar campos dentro de esas fuentes o, en ocasiones, dividir algunos campos individuales en múltiples campos. También es probable que desee sustituir abreviaturas o códigos en los valores que representan (Costello & Blackshear, 2019).

Figura 4

Proceso de extracción, transformación y carga (ETL).



Nota. Microsoft Portal Web.

ETL es el acrónimo que encapsula las tres etapas esenciales en el procesamiento de datos en el contexto de la gestión y análisis de datos. Estas etapas son:

3.2.9.1. Extracción

Durante la etapa de extracción, nuestro enfoque se dirige hacia la conexión con diversas fuentes que pueden incluir bases de datos, archivos planos, servicios web, sistemas en tiempo real y otras fuentes de información. También nos concentraremos en integrar los datos de manera adecuada. La correcta unión de los datos y la comprensión de las consecuencias de cada elección en esta etapa son esenciales, ya que pueden tener un impacto significativo en los datos que se exportarán al final (Costello & Blackshear, 2019).

Esta fase abarca desde la obtención de datos originales hasta la preparación completa de los datos, dejándolos listos para su análisis en la fase de extracción. Finalmente, se

abordarán las ventajas y desventajas de cada enfoque, y comenzaremos a auditar nuestros datos a medida que realizamos las uniones para asegurarnos de obtener los resultados deseados (Costello & Blackshear, 2019).

Palabras claves:

- Fuentes de datos: Las fuentes de datos pueden ser bases de datos, archivos planos, servicios web, sistemas en tiempo real, etc.
- Extracción de Datos: Se refiere a la acción de recuperar datos de estas fuentes. Esto puede implicar la selección de datos relevantes y la aplicación de filtros.
- Integridad de Datos: Asegurar que los datos extraídos sean precisos y completos es fundamental. Esto a menudo implica la validación y limpieza de datos (Costello & Blackshear, 2019).

3.2.9.2. Transformación

Esta fase se hace algunas auditorías simples para ver con qué estamos trabajando, detectar huecos o vacíos en nuestros datos y tomar notas sobre las cosas que queremos cambiar. Después, comenzamos a abordar la noción de "dar forma" a nuestros datos. Este término engloba varios aspectos. Desde una perspectiva general, implica tomar decisiones sobre la estructura de los datos, como determinar si queremos mantener pocas columnas con muchas filas o si preferimos tener muchas columnas con algunas filas (para pivotar o no nuestros datos). También contempla cómo depurar nuestros datos, utilizando una amplia gama de herramientas altamente eficaces para este propósito. Asimismo, consideramos la posibilidad de combinar campos para crear uno nuevo, dividir el contenido de un campo en dos o más campos nuevos y consolidar campos en múltiples registros. (Costello & Blackshear, 2019)

Durante la etapa de transformación, procesamos los datos extraídos para adaptarlos a las necesidades específicas de nuestro negocio. Esto puede implicar la limpieza de datos

para eliminar duplicados y corregir posibles errores, la agregación de información, la conversión de formatos y la aplicación de normas comerciales para enriquecer y preparar los datos para su posterior análisis. (Costello & Blackshear, 2019)

3.2.9.3. Carga

En la etapa de carga, los datos transformados se insertan o cargan en un destino final, que generalmente es un almacén de datos o una base de datos diseñada para admitir análisis y consultas. Los datos se organizan y almacenan de manera estructurada en el destino para facilitar el acceso y el análisis. También podemos crear fuentes de datos intermedias eso podría ser extremadamente útil para probar nuestro proceso y en algunos casos tal vez incluso sean útiles como fuentes de datos por derecho propio (Costello & Blackshear, 2019).

Palabras claves:

- Definición de Esquema: Determinar la estructura de la base de datos o almacén de datos de destino en el que se cargarán los datos.
- Carga de Datos: Mover los datos transformados al destino final, lo que generalmente implica la inserción, actualización o eliminación de registros en una base de datos.
- Gestión de Metadatos: Registrar información sobre la carga de datos, como marcas de tiempo, registros de errores y estadísticas de procesamiento.
- Programación de Carga: Establecer horarios y frecuencias para la ejecución del proceso ETL.
- Programación de Carga: Establecer horarios y frecuencias para la ejecución del proceso ETL (Costello & Blackshear, 2019).

3.2.10. Amazon Web Services (AWS)

Según Amazon Web Services (2025), AWS se posiciona como la plataforma de nube líder y más completa a nivel global, proporcionando una amplia gama de características:

- **Amplia funcionalidad:** AWS es la plataforma de nube más adoptada y completa del mundo. Ofrece más de 200 servicios con diversas funcionalidades, que abarcan desde tecnologías tradicionales como almacenamiento, bases de datos y cómputo, hasta servicios avanzados como inteligencia artificial, aprendizaje automático, análisis, Internet de las cosas (IoT), y tecnologías para desarrolladores móviles. Esta variedad permite a las organizaciones construir e innovar con agilidad en distintos entornos y sectores.
- **Presencia global:** AWS cuenta con una infraestructura global sólida, distribuida en múltiples regiones y zonas de disponibilidad a lo largo del mundo. Esta arquitectura permite a los clientes desplegar aplicaciones en ubicaciones geográficas cercanas a sus usuarios finales, reduciendo la latencia y mejorando la experiencia. Además, brinda redundancia y recuperación ante desastres, lo cual garantiza alta disponibilidad.
- **Comunidad activa y soporte:** AWS posee la comunidad más extensa de clientes activos, incluyendo millones de organizaciones en más de 190 países. Esta comunidad genera un ecosistema colaborativo que permite compartir buenas prácticas, soluciones y soporte técnico a través de foros, capacitaciones y eventos globales. También se encuentra respaldada por una red de socios tecnológicos y de consultoría (AWS Partner Network).
- **Seguridad a nivel empresarial:** La seguridad en AWS es una de sus prioridades principales. Su infraestructura fue diseñada para cumplir con los requisitos más estrictos en materia de protección de datos, confidencialidad y cumplimiento normativo (como GDPR, HIPAA, ISO 27001). Ofrece herramientas avanzadas para

cifrado, control de accesos, auditorías y monitoreo en tiempo real, garantizando un entorno de ejecución confiable y seguro.

- **Innovación constante:** AWS mantiene un enfoque centrado en el cliente, lo que la lleva a desarrollar nuevas funcionalidades a partir de las necesidades reales del mercado. Cada año lanza cientos de mejoras y servicios nuevos, especialmente en campos como la analítica, la inteligencia artificial y el cómputo sin servidor. Esto permite que empresas de todos los tamaños puedan adoptar tecnología de vanguardia de forma escalable y económica (Amazon Web Services, 2025).

Principios claves:

- Nube y computación en la nube: La nube se refiere a la entrega de servicios de Tecnologías de la Información a través de Internet. Los recursos, como servidores, almacenamiento, bases de datos y aplicaciones se proporcionan a pedido y se administran de forma remota en centros de datos de proveedores de servicios en la nube, como AWS. Tiene muchas ventajas entre ellas: escalabilidad, flexibilidad, costo-eficiencia, alta disponibilidad y capacidad de innovación son algunas de las ventajas clave de la computación en la nube (Amazon Web Services, 2025).
- Según Amazon Web Services (2025), la infraestructura y recursos fundamentales que opera en múltiples regiones geográficas en todo el mundo, cada una con múltiples zonas de disponibilidad. Le permite la redundancia y la alta disponibilidad.

Entre sus servicios claves más relevantes tenemos:

- ◊ **Amazon EC2 (Elastic Compute Cloud):** Servicio que proporciona capacidad de cómputo escalable en la nube. Permite crear y administrar máquinas virtuales (instancias) configurables según las necesidades del usuario. Es ideal para ejecutar aplicaciones personalizadas, servidores backend o entornos de procesamiento de datos.
- ◊ **Amazon S3 (Simple Storage Service):** Servicio de almacenamiento de objetos que proporciona escalabilidad, disponibilidad y seguridad. Es utilizado para

almacenar archivos como documentos, registros y respaldos, y permite el acceso desde cualquier ubicación mediante interfaces web o API.

- ◊ **Amazon RDS (Relational Database Service):** Servicio que facilita la configuración, operación y escalado de bases de datos relacionales en la nube. RDS admite motores como MySQL, PostgreSQL y SQL Server, y automatiza tareas como actualizaciones, respaldos y recuperación ante fallos.
- ◊ **AWS Lambda:** Servicio de cómputo sin servidor que permite ejecutar código en respuesta a eventos sin necesidad de aprovisionar o administrar servidores. Se cobra únicamente por el tiempo de ejecución del código, lo cual lo hace eficiente para tareas puntuales y automatizadas.
- ◊ **Amazon VPC (Virtual Private Cloud):** Permite crear redes virtuales privadas aisladas dentro de AWS, con control total sobre el direccionamiento IP, subredes, tablas de rutas y gateways. Es esencial para garantizar la seguridad y el control del entorno de red en aplicaciones empresariales.
- ◊ **Amazon SNS (Simple Notification Service) y Amazon SQS (Simple Queue Service):** Servicios de mensajería que permiten desacoplar microservicios. SNS permite enviar notificaciones a múltiples destinatarios de forma simultánea (modelo de publicación/suscripción), mientras que SQS permite almacenar y procesar mensajes de forma asíncrona mediante colas.
- ◊ **Amazon IAM (Identity and Access Management):** Servicio para gestionar de forma segura el acceso a los recursos de AWS mediante políticas, usuarios, roles y grupos. IAM permite aplicar el principio de mínimo privilegio y controlar quién puede hacer qué dentro de la infraestructura (Amazon Web Services, 2024).

3.2.11. Pandas

Pandas es una biblioteca de código abierto para análisis y manipulación de datos construida sobre Python. Ofrece estructuras de datos de alto rendimiento, como DataFrames y Series, que facilitan el manejo de grandes conjuntos de datos para tareas como la limpieza, transformación y exploración de datos. Su funcionalidad robusta permite a los científicos de datos trabajar eficazmente con datos estructurados y no estructurados, convirtiéndola en una herramienta clave en el ecosistema de ciencia de datos en Python. (Pandas Development Team, 2024)

3.2.12. Selenium

Selenium en Python es una herramienta para automatizar navegadores web, utilizada comúnmente en pruebas de aplicaciones web y para automatizar tareas repetitivas como el web scraping y la navegación programática. Selenium permite que los scripts de Python interactúen con elementos de una página web, imitando acciones del usuario como hacer clic, escribir y navegar entre páginas. Es compatible con múltiples navegadores (por ejemplo, Chrome, Firefox, Safari y Edge) y utiliza controladores específicos de cada navegador para facilitar la interacción. Con Selenium WebDriver, los desarrolladores pueden crear pruebas de extremo a extremo, simulando interacciones complejas de usuarios en distintos entornos de navegadores, lo que es ideal para probar aplicaciones en tiempo real. (Bhagya, 2023)

La API de Selenium para Python es versátil y brinda soporte completo para manejar contenido dinámico y gestionar elementos del navegador mediante funciones como localizar elementos por ID, clase o XPath. Selenium también puede realizar tareas avanzadas como cambiar entre ventanas, manejar alertas y desplazarse por la página. Una fortaleza clave es su soporte para múltiples estrategias de espera (esperas explícitas e implícitas), lo cual permite que los scripts esperen a que los elementos se carguen antes de realizar

acciones, lo cual es esencial para manejar contenido web asincrónico. (The Selenium Project, 2023)

Capítulo 4

Desarrollo del sistema

Tal como se mencionó anteriormente, el desarrollo del sistema se llevó a cabo empleando la metodología ágil Scrum. Esta metodología se distingue por organizar las funcionalidades en **historias de usuario**, que a su vez se planifican en sprints con un conjunto específico de tareas priorizadas en el backlog.

Cada sprint cuenta con un tiempo definido de ejecución y produce un entregable funcional, lo que permite obtener un producto en evolución progresiva hasta alcanzar la solución final.

4.1. Metodología del proyecto

4.1.1. Aplicación de Scrum

En el proyecto se establecieron los siguientes elementos:

- **Roles:** Product Owner, Scrum Master y Development Team.
- **Artefactos:** Product Backlog, Sprint Backlog e Incremento.
- **Eventos:** Planificación del sprint, reuniones diarias, revisión del sprint y retrospectiva.

4.2. Historias de usuario por Sprint

Historia de usuario 1

Sprint: Sprint 1

Rol: Desarrollador

Objetivo: Como desarrolladores, queremos comprender los requerimientos, las áreas involucradas y el soporte necesario para planificar eficientemente el desarrollo.

Beneficio: Obtener los requerimientos y profundizar en la información de las áreas involucradas en el desarrollo.

Backlog:

- Reuniones realizadas con stakeholders.
- Documentar información recopilada de las reuniones realizadas.

Duración: 1 semana

Tabla 3
Requerimientos funcionales

ID	Requerimiento	Prioridad	Área	Observaciones
RQ01	Extracción de datos desde portal web interno: Acepta	Alta	TI	Necesario para iniciar el procesamiento automatizado con RPA.
RQ02	Extracción de datos desde sistema interno RMS y SAP	Alta	TI	Store Procedure personalizados para cada sistema (RMS, SAP).
RQ03	Unificación y limpieza de datos de múltiples plataformas	Alta	TI	Debe permitir análisis homogéneo y consistente.
RQ04	Notificación diaria por correo con alertas del proceso	Alta	TI/Contabilidad	Requiere configuración de correos HTML y destinatarios.
RQ05	Despliegue del sistema de gestión en Amazon Web Services (AWS)	Alta	TI	Infraestructura en nube con acceso y seguridad.
RQ06	Pruebas de funcionamiento del sistema para validar alertas y escenarios	Alta	TI	—

Nota. Elaboración propia.

Historia de usuario 2

Sprint: Sprint 2,3,4,5

Rol: Desarrollador

Objetivo: Como desarrolladores, queremos extraer los datos de los documentos contables de la plataforma Acepta, RMS y SAP.

Beneficio: Facilitar el acceso y la gestión de los datos, para obtener una visión completa de la información contable extraída.

Backlog:

- Desarrollar un RPA para extraer la data del portal web de Acepta.
- Desarrollar un SP en SQL para obtener los datos de los documentos contables emitidos por el sistema RMS.
- Desarrollar un SP en SQL para obtener los datos de los documentos contables guardados en el sistema SAP.

Duración: 4 semanas

Historia de usuario 3

Sprint: Sprint 6,7,8,9

Rol: Desarrollador

Objetivo: Como desarrolladores, queremos unificar y limpiar los datos contables de Acepta, RMS y SAP para analizarlos de forma homogénea y confiable.

Beneficio: Unificar y depurar datos contables de distintas plataformas para garantizar consistencia y optimizar su análisis.

Backlog:

- Convertir los archivos .csv y .xls en un dataframe unificado y depurado.
- Convertir los .sql en un dataframe unificado y depurado.

Duración: 4 semanas

Historia de usuario 4

Sprint: Sprint 10

Rol: Miembros del área de TI y Contabilidad.

Objetivo: Como miembros del área de TI y Contabilidad, quiero recibir alertas por correo diariamente al finalizar el proceso de verificación para conocer los documentos con errores.

Beneficio: Obtener los documentos contables que no figuren en SAP, RMS Y Acepta.

Backlog:

- Automatizar el envío de correos en formato html a los destinatarios ya determinados con la información procesada.

Duración: 1 semana

Historia de usuario 5

Sprint: Sprint 11

Rol: Product Owner

Objetivo: Como Product Owner, quiero desplegar la implementación del Sistema de Gestión e integración de información contable y financiera en Amazon Web Service.

Beneficio: Tener el sistema desplegado en AWS a disponibilidad de la empresa y así resguardar dicha información.

Backlog:

- Crear instancias.
- Configurar puertos e IP.
- Configuración de tarea programada.

Duración: 1 semana

Historia de usuario 6

Sprint: Sprint 12

Rol: Product Owner

Objetivo: Como Product Owner, quiero testear el funcionamiento del sistema, para validar el proceso de matching de documentos contables, mapear los escenarios posibles y revisar como llegarían las alertas a los correos.

Beneficio: Obtener un sistema que tenga los requerimientos necesarios que se plantearon al iniciar toda la implementación.

Backlog:

- Realizar pruebas integrales del sistema.

Duración: 1 semana

Tabla 4*Historias de usuario, backlog y relación con requerimientos y sprints*

ID HU	Historia de usuario	Req. relacionado	Backlog	Sprint
HU01	Como desarrollador, quiero comprender los requerimientos, las áreas involucradas y el soporte necesario para planificar eficientemente el desarrollo.	–	Reuniones realizadas con stakeholders. Documentar información recopilada de las reuniones realizadas.	Sprint 1
HU02	Como desarrollador, quiero extraer los datos de los documentos contables de la plataforma Acepta, RMS y SAP.	RQ01, RQ02	Desarrollar un RPA para extraer la data del portal web de Acepta. Desarrollar un SP en SQL para obtener los datos de los documentos contables emitidos por el sistema RMS. Desarrollar un SP en SQL para obtener los datos de los documentos contables guardados en el sistema SAP.	Sprint 2–5
HU03	Como desarrollador, quiero unificar y limpiar los datos contables de Acepta, RMS y SAP para analizarlos de forma homogénea y confiable.	RQ03	Archivos convertidos en dataframes unificados. Datos depurados sin duplicados.	Sprint 6–9
HU04	Como miembros del área de TI y Contabilidad, quiero recibir alertas por correo diariamente al finalizar el proceso de verificación para conocer los documentos con errores.	RQ04	Envío automatizado de correos HTML. Contenido claro y filtrado por error. Entregas diarias.	Sprint 10
HU05	Como Product Owner, quiero desplegar el sistema de gestión e integración contable y financiera en AWS para asegurar su disponibilidad y resguardo.	RQ05	Instancias creadas en AWS. Puertos e IP configurados. Tarea programada funcional.	Sprint 11
HU06	Como Product Owner, se requiere testear el funcionamiento del sistema para validar el proceso de matching de documentos contables, mapear los escenarios posibles y revisar cómo llegarían las alertas a los correos.	RQ06	Pruebas ejecutadas con reporte. Verificación de alertas recibidas. Feedback documentado.	Sprint 12

Nota. Elaboración propia.

4.3. Automatización de descarga de documentos del portal ACEPTA usando RPA.

Para poder desarrollar el Sistema se necesita implementar un Bot para descargar los reportes del portal web ACEPTA para lo cual usaremos como herramienta python y las librerías correspondientes para poder automatizar las descargas simulando que fuera un usuario del portal web acepta.

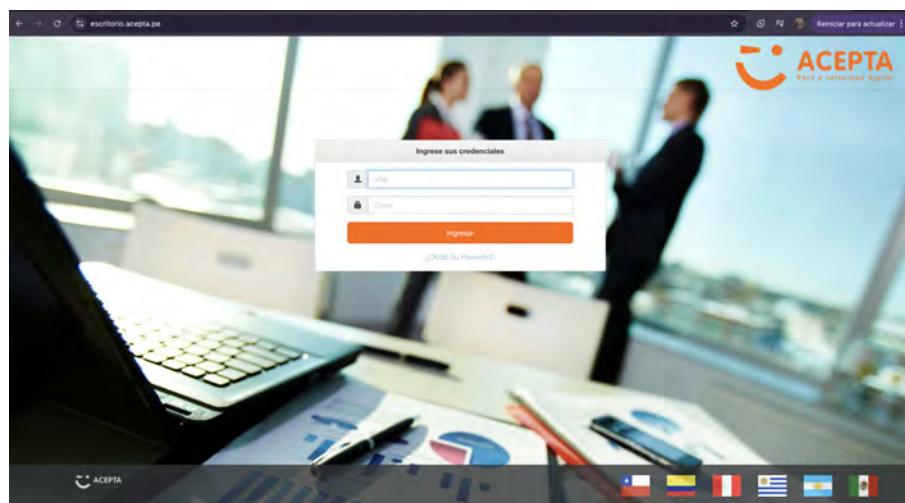
Primero identificaremos los pasos a seguir para descargar la información de los documentos contables.

1. Abrir un navegador, en este caso utilizaremos el navegador Google Chrome e ingresamos la url: <https://escritorio.acepta.pe> .

Código 1. Inicialización del navegador con Selenium

```
1     driver = webdriver.Chrome()  
2     driver.get(ACEPTA_URL)  
3     driver.maximize_window()  
4     time.sleep(2)
```

Figura 5
Portal de Acepta



Nota. Acepta Portal Web.

2. Ingresar las credenciales: usuario y contraseña, iniciar sesión y esperar cinco segundo a que cargue la plataforma web.

Código 2. Loguearse en la plataforma

```
1 # usuario
2 user = driver.find_element("xpath", '//*[@id="loginrut"]')
3 user.send_keys(usuario)
4
5 # password
6 password = driver.find_element(
7     "xpath", '//*[@id="container_login_form"]/div/div[2]/form/fieldset/div[2]/div[2]/div[2]/div[1]',
8     )
9 password.send_keys(contrasenia)
10
11 # iniciar sesion
12 mBox = driver.find_element(
13     "xpath", '//*[@id="container_login_form"]/div/div[2]/form/fieldset/div[2]/div[3]/input',
14     ).click()
```

Figura 6
Login de Acepta



Nota. Acepta Portal Web.

3. Dar clic en el botón "Aceptar" del modal de información. El modal informativo, solo aparece cuando la plataforma web acepta quiera mandar algún comunicado a sus usuarios. En este caso/fecha no se pudo capturar dicho modal porque no había ninguno vigente.

Código 3. Modal de información

```
1 # clic en aceptar en el mensaje de alerta
2 time.sleep(5)
3 try:
4     mBox = driver.find_element("xpath", "//button[@id='btn_close_aviso_pantalla']"
5                                →  "").click()
5 except:
6     pass
```

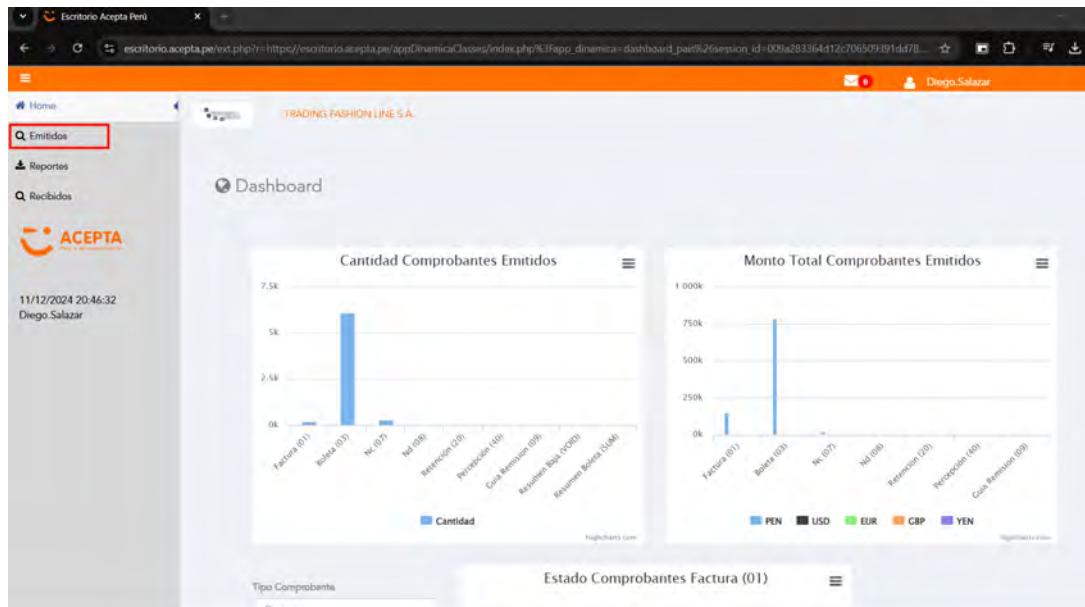
4. Ir al menú de "emitidos" y luego a búsqueda avanzadas.

Código 4. Menú de emitidos y búsqueda avanzada

```
1 # emitidos
2 mBox = driver.find_element("xpath", '//*[@id="badge_185"]').click()
3
4 # busqueda avanzada
5 mBox = driver.find_element("xpath", '//*[@id="form_params"]/div[5]/a').click()
6 time.sleep(2)
```

Figura 7

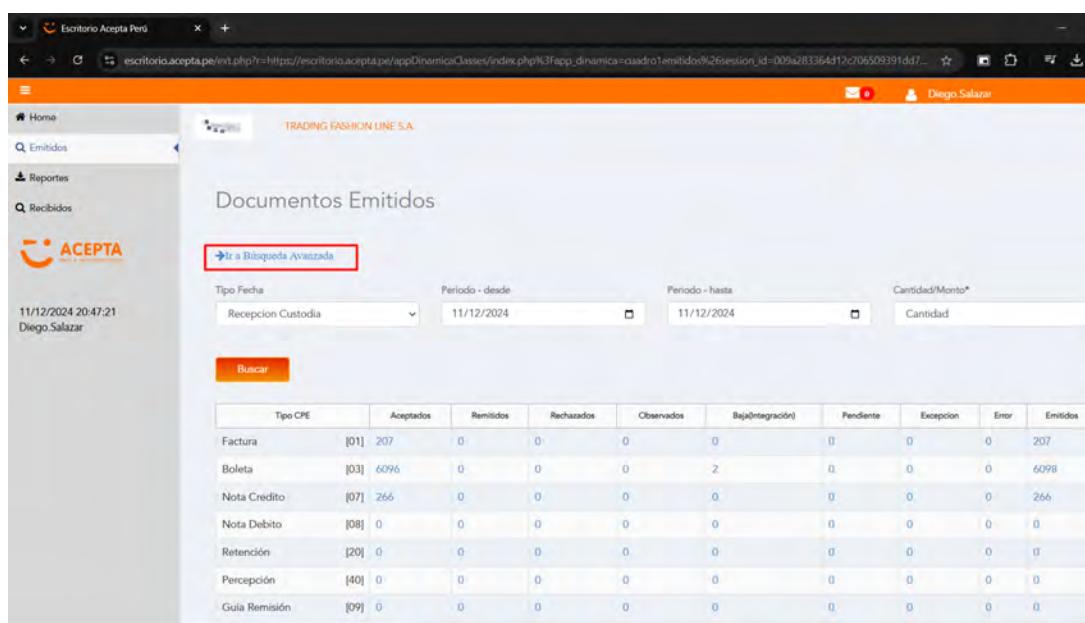
Menú emitidos



Nota. Acepta Portal Web.

Figura 8

Búsqueda avanzada en documentos emitidos



Nota. Acepta Portal Web.

5. Seleccionar el menú desplegable “Tipo fecha” y elegir “Emisión”, luego Seleccionar el menú “Periodo – desde” y elegir 7 días atrás con referencia a la fecha actual, para

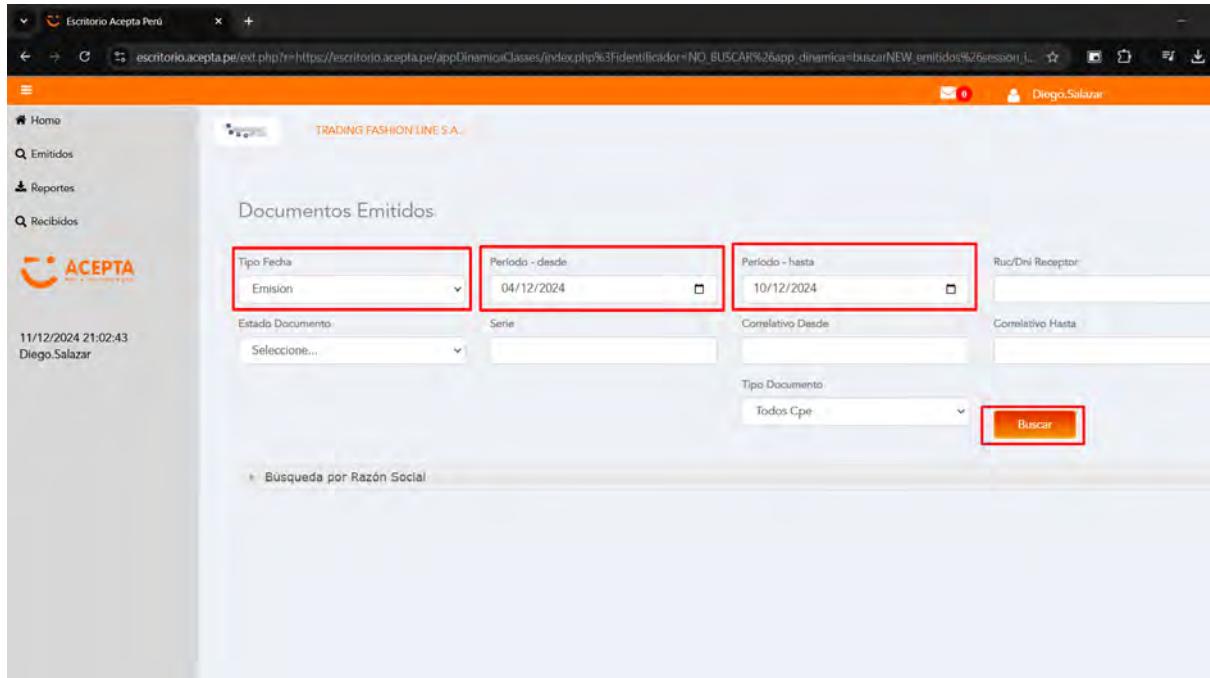
después Seleccionar el menú “Periodo – hasta” y elegir 1 día atrás con referencia de hoy. Y finalmente clic en el botón “Buscar”.

Código 5. Menús desplegables dentro de emitidos

```
1 # tipofecha = Emision
2 mBox = driver.find_element("xpath", '//*[@id="TIPO_FECHA"]/option[2]').click()
3
4 fIni = date.today() - timedelta(days=7)
5 fFin = date.today() - timedelta(days=1)
6
7 # periodoDesde-- 7 dias menos
8 fechaIni = driver.find_element("xpath", "//input[@name='FSTART']")
9 fechaIni.send_keys(fIni.strftime("%m%d%Y")) #("%d%m%Y")
10
11 # periodoHasta--1 dia menos
12 fechaFin = driver.find_element("xpath", "//input[@name='FEND']")
13 fechaFin.send_keys(fFin.strftime("%m%d%Y")) #("%d%m%Y")
14
15 # boton Buscar
16 buscar = driver.find_element("xpath", "//input[@name='buscar2']").click()
17 time.sleep(8)
```

Figura 9

Menús desplegables en documentos emitidos



Nota. Acepta Portal Web.

6. Esperar que botón "exportar" este disponible y luego hacer clic botón exporta.

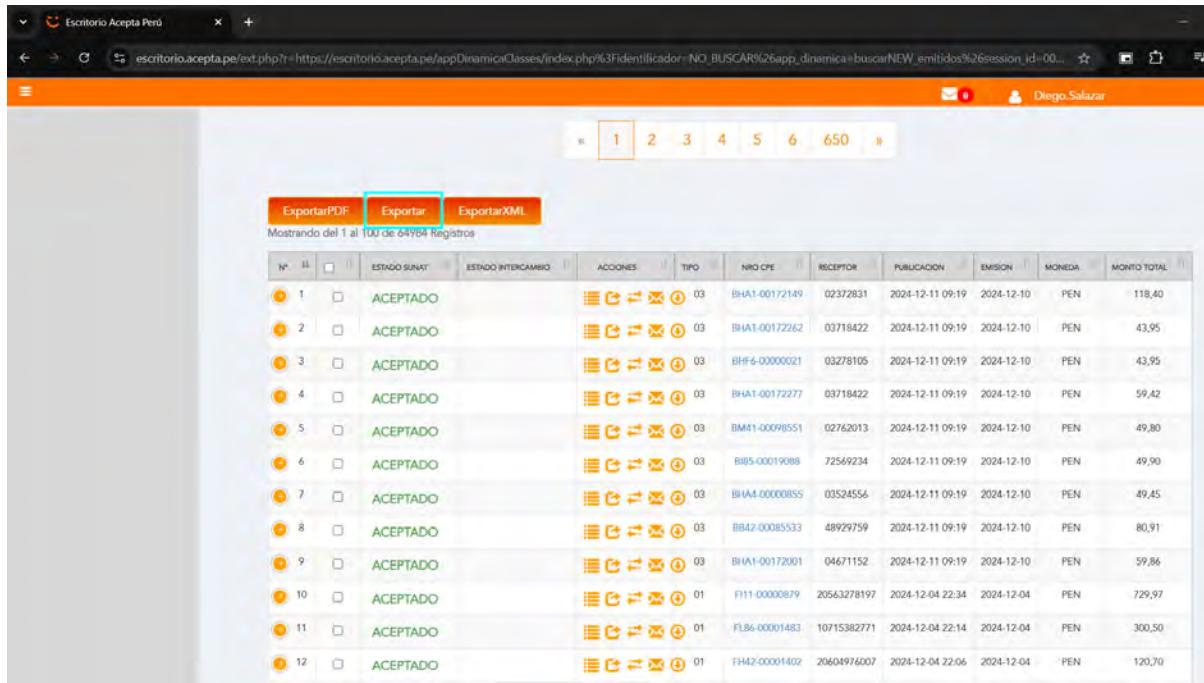
Código 6. Exportar emitidos

```

1 # esperar que boton "exportar" este disponible
2 # clic boton exportar
3 ex = False
4 while ex == False:
5     try:
6         expor = driver.find_element(
7             "xpath", "(//input[@name='Exportar'])[1]")
8         expor.click()
9         ex = True
10    except:
11        pass

```

Figura 10
Exportar documentos emitidos



The screenshot shows a table with 12 rows of data. Each row contains a small orange circle with a number (1-12), a checkbox, the text 'ACEPTADO', and a series of icons. The columns are labeled: N°, ESTADO SUNAT, ESTADO INTERCAMBIO, ACCIONES, TIPO, NRO CPE, RECEPTOR, PUBLICACION, EMISION, MONEDA, and MONTO TOTAL. The 'Exportar' button at the top of the table is highlighted with a blue box.

N°	ESTADO SUNAT	ESTADO INTERCAMBIO	ACCIONES	TIPO	NRO CPE	RECEPTOR	PUBLICACION	EMISION	MONEDA	MONTO TOTAL
1	ACEPTADO			03	BHA1-00172149	02372831	2024-12-11 09:19	2024-12-10	PEN	118,40
2	ACEPTADO			03	BHA1-00172262	03718422	2024-12-11 09:19	2024-12-10	PEN	43,95
3	ACEPTADO			03	BHF6-00000021	03278105	2024-12-11 09:19	2024-12-10	PEN	43,95
4	ACEPTADO			03	BHA1-00172277	03718422	2024-12-11 09:19	2024-12-10	PEN	59,42
5	ACEPTADO			03	BMH1-00098551	02762013	2024-12-11 09:19	2024-12-10	PEN	49,80
6	ACEPTADO			03	BB5-000019088	72569234	2024-12-11 09:19	2024-12-10	PEN	49,90
7	ACEPTADO			03	BHA1-00000055	03524556	2024-12-11 09:19	2024-12-10	PEN	49,45
8	ACEPTADO			03	BB42-00085533	48929759	2024-12-11 09:19	2024-12-10	PEN	80,91
9	ACEPTADO			03	BHA1-00172001	04671152	2024-12-11 09:19	2024-12-10	PEN	59,86
10	ACEPTADO			01	FI11-00000879	20563278197	2024-12-04 22:34	2024-12-04	PEN	729,97
11	ACEPTADO			01	FLB6-00001483	10715382771	2024-12-04 22:14	2024-12-04	PEN	300,50
12	ACEPTADO			01	FH42-00001402	20604976007	2024-12-04 22:06	2024-12-04	PEN	120,70

Nota. Acepta Portal Web.

7. Clic en "Aceptar" dentro del mensaje de tipo modal, que indica que la exportación se está realizando.

Código 7. Modal de confirmación

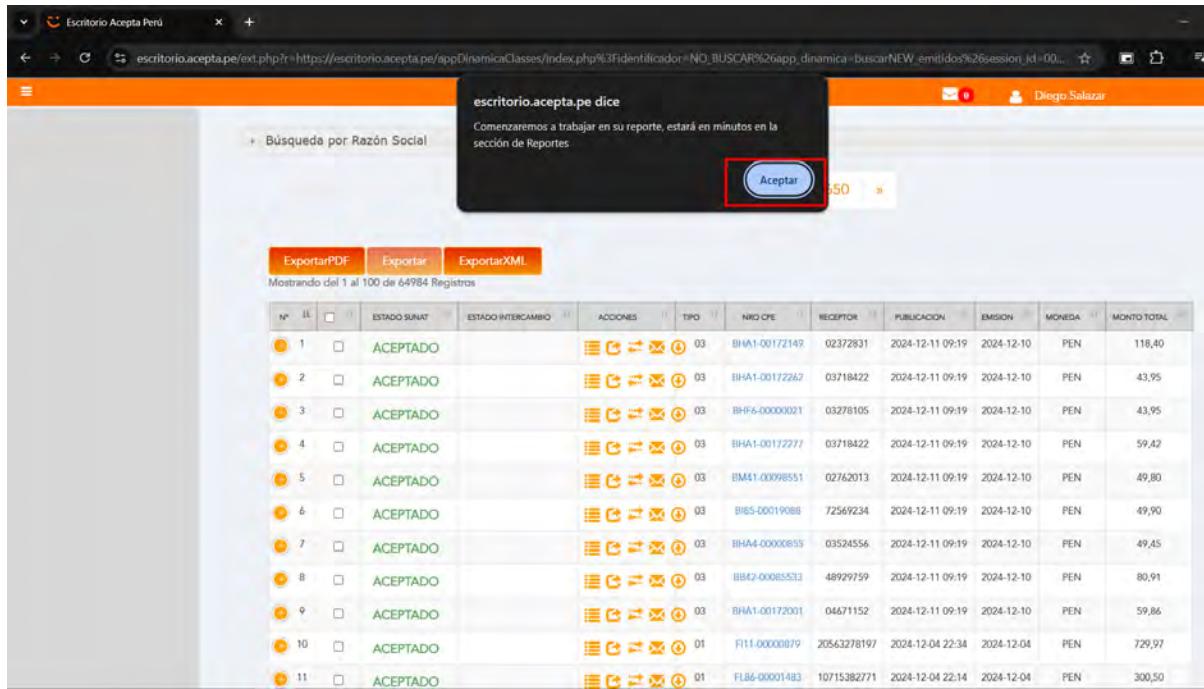
```

1  # enter
2
3  WebDriverWait(driver, 10).until(EC.alert_is_present())
4
5  driver.switch_to.alert.accept()

```

Figura 11

Modal de confirmación de la exportación



Nota. Acepta Portal Web.

8. Ir al menú lateral y elegir “reportes”.

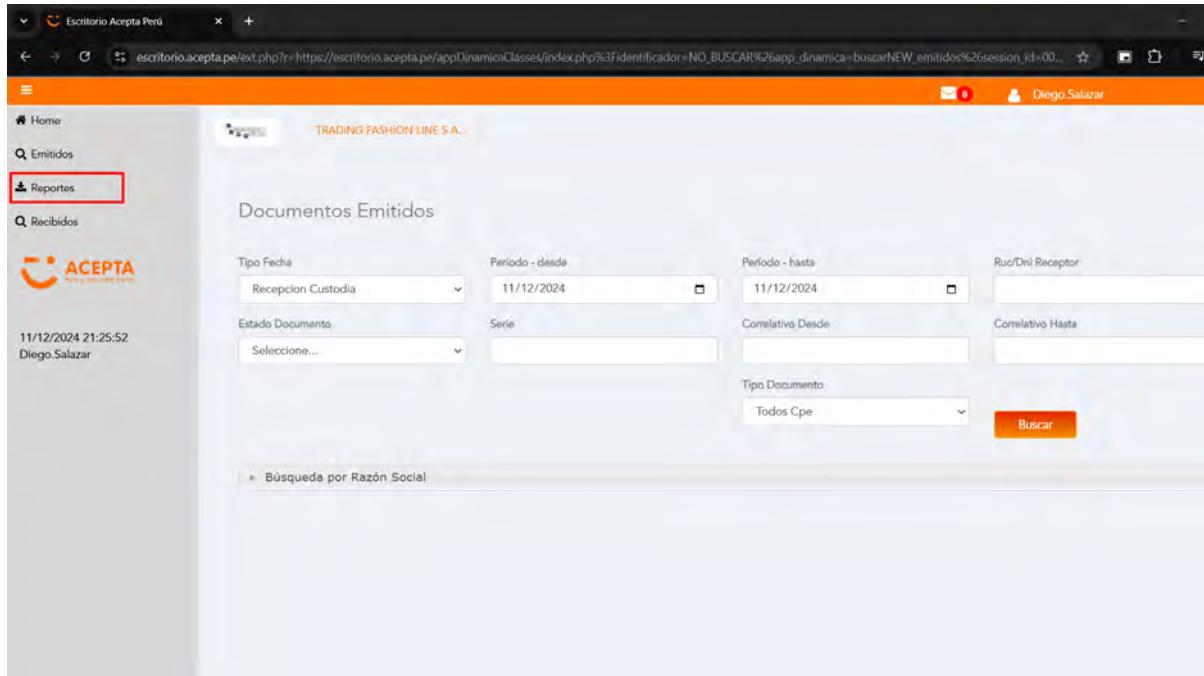
Código 8. Ir a reportes

```

1 # ir a reportes
2 reportes = driver.find_element("xpath", '//*[@id="badge_193"]').click()
3 time.sleep(180)

```

Figura 12
Ir al menú de reportes



Nota. Acepta Portal Web.

9. Dar clic en el botón "Descargar", cuando este disponible, en el último item del listado de reportes. La carpeta destino para descargar, en este caso será una ruta del servidor:

```
r"\\SRVRMS01\\VentasRMS" + "\\CruceAcepta_" + fecha + ".zip"
```

Código 9. Descargar reportes

```

1 # descargar reportes
2 btemp = False
3
4 while btemp == False:
5     try:
6         descg = driver.find_element(
7             "xpath", "//a[@id='1Descargargrilla_reportesNEW']")
8         .get_attribute("href")
9

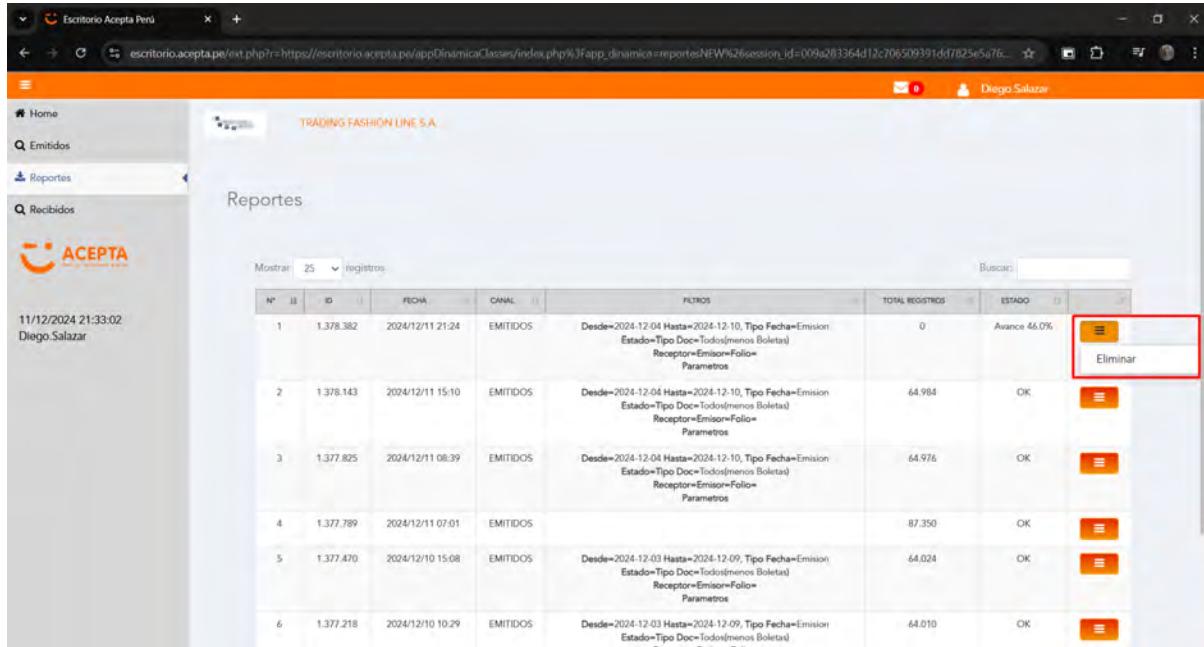
```

```
10     fecha = date.today() - timedelta(days=1)
11     fecha = fecha.strftime("%Y%m%d")
12
13     ruta = r"\191.1.64.110\VentasRMS\CruceAcepta_" + fecha + ".zip"
14     ruta2 = r"\191.1.64.110\VentasRMS\CruceAcepta_" + fecha + ".csv"
15     ruta3 = r"\191.1.64.110\VentasRMS\CruceAcepta_" + fecha + ".xls"
16
17     if os.path.exists(ruta):
18         os.remove(ruta)
19
20     if os.path.exists(ruta2):
21         os.remove(ruta2)
22
23     if os.path.exists(ruta3):
24         os.remove(ruta3)
25
26     wget.download(
27         descg, r"\191.1.64.110\VentasRMS" + "\CruceAcepta_" + fecha +
28         ".zip")
29     btemp = True
30
31     except Exception as e:
32         print(e)
33         time.sleep(60)
34     pass
```

CAPÍTULO 4. Desarrollo del sistema

Figura 13

Espera de la habilitación del botón de descarga



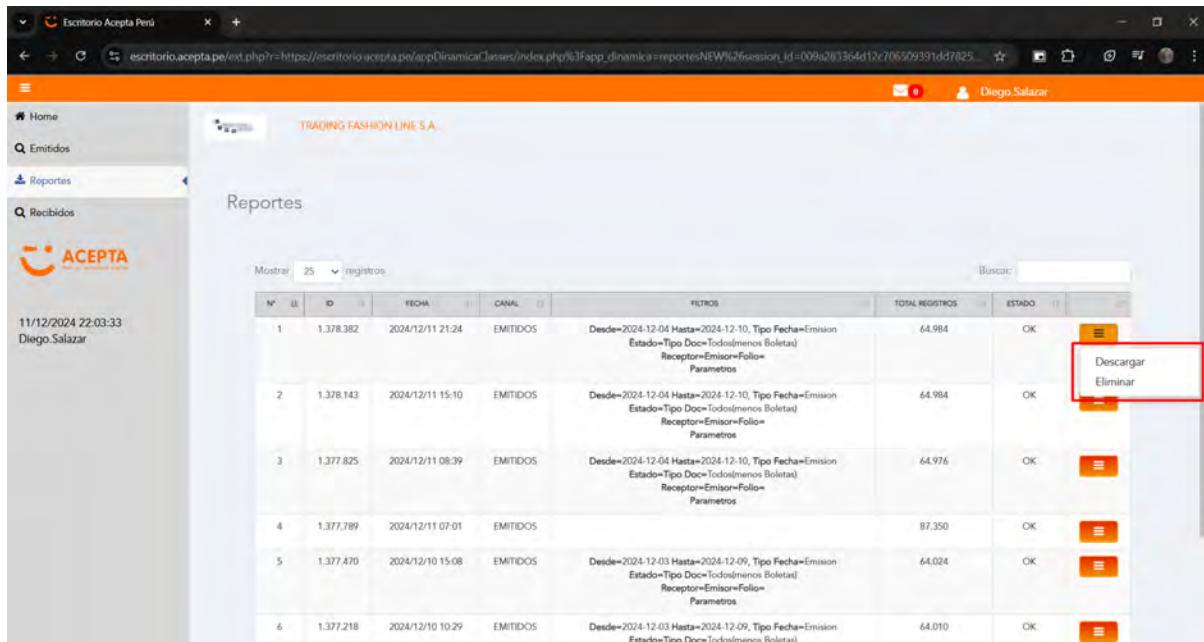
The screenshot shows a web browser window for 'Escritorio Acepta Perú'. The URL is 'escritorio.acepta.pe/ext.php?r=https://escritorio.acepta.pe/appDinamicaObras/index.php%3fapp_dinamica=reporteNEW%26session_id=09a201364d12c706509391d77025e5a7b...'. The page title is 'TRADING FASHION LINE S.A.'. The left sidebar has 'Home', 'Emitidos', 'Reportes', and 'Recibidos'. The main area is titled 'Reportes' and shows a table with 6 rows. The table has columns: N°, ID, FECHA, CANAL, FILTROS, TOTAL REGISTROS, ESTADO, and two buttons: 'Eliminar' and 'Descargar'. The 'Eliminar' button is highlighted with a red box. The 'Descargar' button is not visible in this view.

N°	ID	FECHA	CANAL	FILTROS	TOTAL REGISTROS	ESTADO	Eliminar	Descargar
1	1.378.382	2024/12/11 21:24	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	0	Avance 46.0%		
2	1.378.143	2024/12/11 15:10	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.984	OK		
3	1.377.825	2024/12/11 08:39	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.976	OK		
4	1.377.789	2024/12/11 07:01	EMITIDOS		87.350	OK		
5	1.377.470	2024/12/10 15:08	EMITIDOS	Desde=2024-12-03 Hasta=2024-12-09, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.024	OK		
6	1.377.218	2024/12/10 10:29	EMITIDOS	Desde=2024-12-03 Hasta=2024-12-09, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta)	64.010	OK		

Nota. Acepta Portal Web.

Figura 14

Botón habilitado de descarga



The screenshot shows a web browser window for 'Escritorio Acepta Perú'. The URL is 'escritorio.acepta.pe/ext.php?r=https://escritorio.acepta.pe/appDinamicaObras/index.php%3fapp_dinamica=reporteNEW%26session_id=09a201364d12c706509391d77025e5a7b...'. The page title is 'TRADING FASHION LINE S.A.'. The left sidebar has 'Home', 'Emitidos', 'Reportes', and 'Recibidos'. The main area is titled 'Reportes' and shows a table with 6 rows. The table has columns: N°, ID, FECHA, CANAL, FILTROS, TOTAL REGISTROS, ESTADO, and two buttons: 'Eliminar' and 'Descargar'. The 'Descargar' button is highlighted with a red box. The 'Eliminar' button is not visible in this view.

N°	ID	FECHA	CANAL	FILTROS	TOTAL REGISTROS	ESTADO	Eliminar	Descargar
1	1.378.382	2024/12/11 21:24	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.984	OK		
2	1.378.143	2024/12/11 15:10	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.984	OK		
3	1.377.825	2024/12/11 08:39	EMITIDOS	Desde=2024-12-04 Hasta=2024-12-10, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.976	OK		
4	1.377.789	2024/12/11 07:01	EMITIDOS		87.350	OK		
5	1.377.470	2024/12/10 15:08	EMITIDOS	Desde=2024-12-03 Hasta=2024-12-09, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta) Receptor=Emisor=Folio=Parametros	64.024	OK		
6	1.377.218	2024/12/10 10:29	EMITIDOS	Desde=2024-12-03 Hasta=2024-12-09, Tipo Fecha=Emission Estado=Tipo Doc=Todos(menos Boleta)	64.010	OK		

Nota. Acepta Portal Web.

10. Extraer zip y renombrar archivo xls

Código 10. Extraer zip, renombrar .xls

```
1 # extraer zip y renombrar archivo xls
2 driver.close()
3
4 fecha = date.today() - timedelta(days=1)
5 fecha = fecha.strftime("%Y%m%d")
6
7 archivo_zip = zipfile.ZipFile(r"\191.1.64.110\VentasRMS\\CruceAcepta_" + fecha
8     + ".zip")
9
10 nameOriginal = archivo_zip.namelist()
11 tempStr = "".join(nameOriginal)
12
13 if 'xls' in nameOriginal[0]:
14     NewName = r"\191.1.64.110\VentasRMS" + "\\CruceAcepta_" + fecha + ".xls"
15 else:
16     NewName = r"\191.1.64.110\VentasRMS" + "\\CruceAcepta_" + fecha + ".csv"
17
18 ArchivoExcel = r"\191.1.64.110\VentasRMS" + "\\\" + tempStr
19
20 os.rename(ArchivoExcel, NewName)
```

4.4. Extracción y transformación de datos de los documentos del portal ACEPTA.

En esta parte se hace una limpieza de los datos, el cual incluye la eliminación de las columnas que no son relevantes para el resultado final y luego se hace una transformación en la columna ”nro_cpe_” el cual debe tener la misma estructura de los demás data frames a procesar.

Código 11. Transformación de la data de los documentos del portal ACEPTA

```
1 if 'xls' in nameOriginal[0]:  
2     df1=pd.read_excel(NewName)  
3 else:  
4     df1=pd.read_csv(NewName)  
5  
6 df1.drop(["receptor",  
7           "razon_social_receptor",  
8           "moneda",  
9           "isc",  
10          "ivap",  
11          "operacionesexoneradas",  
12          "operacionesinafectas",  
13          "operacionesgratis",  
14          "descuentoglobal",  
15          "otroscargos",  
16          "otrostributos",  
17          "icbper",  
18          "resumen_boleta",  
19          "uri",  
20          "referencias",  
21          "mensaje_sii",  
22          "motivo_observacion",  
23          ],
```

```
24     axis=1,  
25     inplace=True,  
26     )  
27 df1 = df1[df1['tipo'] != 20]  
28 df1['nro_cpe_'] = df1.apply(lambda row: '0'+str(row['tipo'])+'-' + row['nro_cpe'], axis=1)  
29  
30 return df1
```

4.5. Extracción y transformación de datos de los documentos de la Base de datos del sistema Retail Management System.

En esta parte configuramos la conexión, y nos conectamos al servidor de RMS, luego seteamos las fechas: inicio y fin. Para luego ejecutar un store procedure con los parámetros correspondientes fecha y número de tienda, en número de tienda se puso el numeral 9999 para que liste todas las tiendas. Para finalmente ser recuperado/leído en un dataframe.

Código 12. Transformación de datos de los documentos de la Base de datos del sistema RMS.

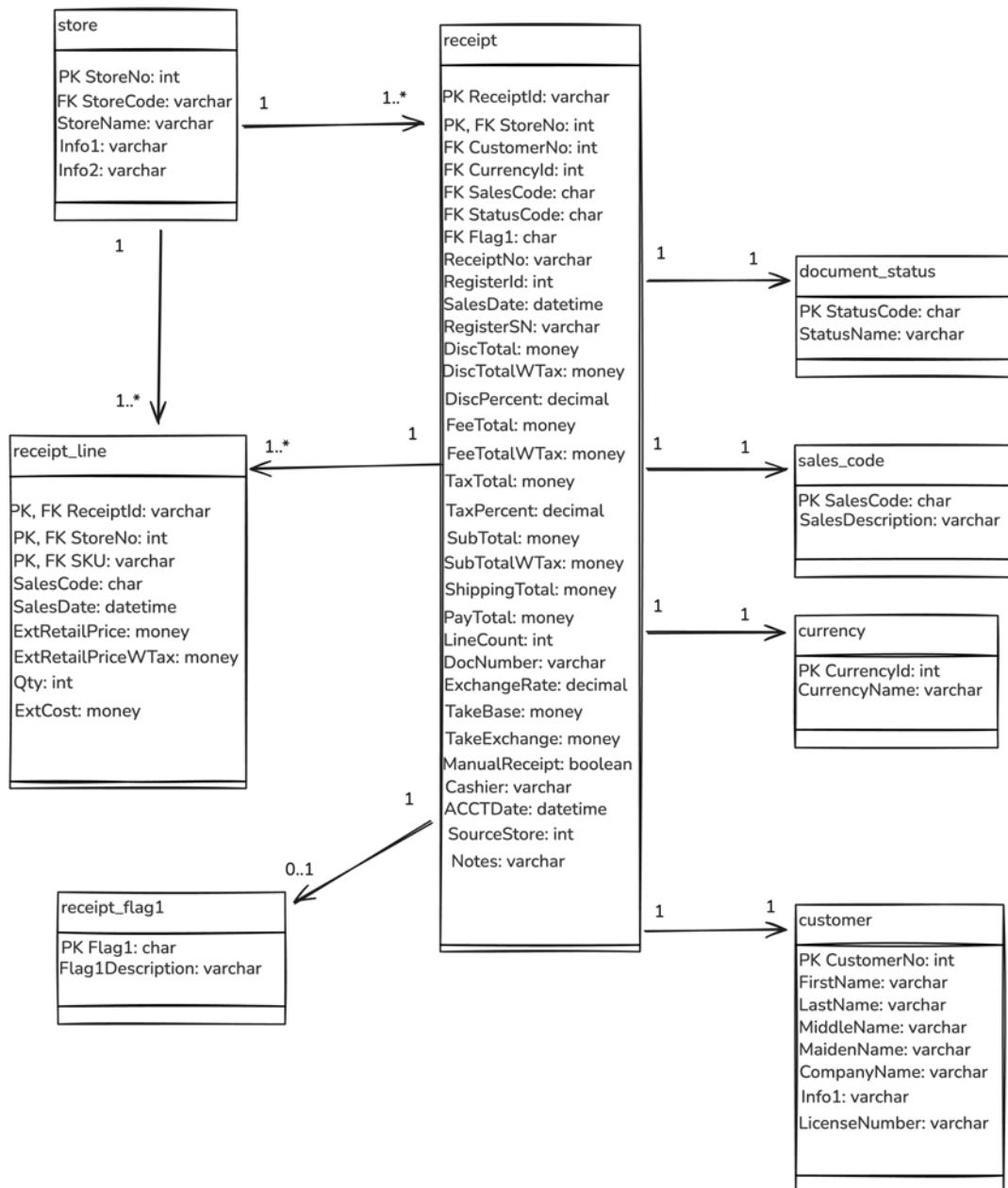
```
1 def consultaRMS_BD():
2     server = IP_SERVER_RMS
3     database = DATABASE_RMS
4     username = USERNAME_DATABASE_RMS
5     password = PASSWORD_DATABASE_RMS
6
7     # cadena de conexión
8     conn = pyodbc.connect(
9         f"DRIVER={{SQL Server Native Client
10             \r 11.0}};SERVER={server};DATABASE={database};UID={username};PWD={password}"
11     )
12     cursor = conn.cursor()
13
14     # setear fecha inicio/fin un dia anterior
15     fIni = datetime.now() - timedelta(days=7)
16     fIni = fIni.replace(hour=0, minute=0, second=0).strftime("%Y-%m-%d %H:%M:%S")
17     fFin = datetime.now() - timedelta(days=1)
18     fFin = fFin.replace(hour=23, minute=59, second=59).strftime("%Y-%m-%d %H:%M:%S")
19     query = "EXEC Sp_Integration_StoreSummaryCustomerCompany_forDay_ @DateFrom = '{0}'"
20     \r , @DateTo='{1}', @StoreNo={2}"".format(
```

```
19     fIni, fFin, 9999
20 )
21
22 # mostrar dataframe usando lib pandas
23 dfRMS = pd.read_sql_query(query, conn)
24 return dfRMS
25 conn.close()
```

El SP "Sp_Integration_StoreSummaryCustomerCompany_forDay" genera un resumen detallado de las transacciones (ventas, devoluciones, descuentos, etc.) por tienda en un rango de fechas, con desglose por cliente, artículos, márgenes, descuentos, etc. Pasaremos a detallar dicho desarrollo, comenzando por el diagrama de clases:

Figura 15

Diagrama de base de datos que utiliza el SP



Nota. Elaboración propia.

Código 13. Creación del procedimiento almacenado

```

1
2 CREATE PROCEDURE [dbo].[Sp_Integration_StoreSummaryCustomerCompany_forDay]
3     @DateFrom datetime,
4     @DateTo datetime,

```

```

5      @StoreNo int
6  AS
7  BEGIN
8      EXEC [Custm_Topitop_StoreSummaryCustomerCompany_forDay] '20170901 000000',
9          '20170930 235959', 27;
10
11     CREATE TABLE #ttStore (
12
13         StoreNo int,
14
15         StoreCode varchar(10) COLLATE SQL_Latin1_General_CI_AS NULL,
16
17         StoreName varchar(50) COLLATE SQL_Latin1_General_CI_AS NULL,
18
19         StoreInfo1 varchar(50) COLLATE SQL_Latin1_General_CI_AS NULL,
20
21         StoreInfo2 varchar(50) COLLATE SQL_Latin1_General_CI_AS NULL
22
23     );
24
25
26     IF @StoreNo = 9999
27
28         INSERT INTO #ttStore
29
30         SELECT StoreNo, StoreCode, StoreName, Info1, Info2
31
32         FROM store
33
34         WHERE storetype != 'B';
35
36     ELSE
37
38         INSERT INTO #ttStore
39
40         SELECT StoreNo, StoreCode, StoreName, Info1, Info2
41
42         FROM store
43
44         WHERE StoreNo = @StoreNo;
45
46
47     -----
48
49     -- VentaCuponesDescuento (solo ventas)
50
51     -----
52
53     SELECT R. *
54
55     -- INTO #ttRR
56
57     FROM receipt R (NOLOCK)
58
59     INNER JOIN #ttStore ON R.StoreNo = #ttStore.StoreNo
60
61     WHERE R.SalesCode = 'S'
62
63         AND R.SalesDate BETWEEN @DateFrom AND @DateTo
64
65         AND R.STATUSCODE = 'A'

```

```
39      AND R.Version != 'SYSVENT';

40

41  SELECT
42
43      R.StoreNo,
44
45      R.ReceiptId,
46
47      ISNULL(SUM(ABS(RL.ExtRetailPriceWTax)), 0) AS DiscAmountLine
48
49  INTO #RECEIPT_RETURN
50
51  FROM #ttRR R
52
53  INNER JOIN RECEIPT_LINE RL (NOLOCK) ON R.ReceiptId = RL.ReceiptId AND R.StoreNo =
54
55      ↪ RL.StoreNo
56
57  WHERE R.SalesCode = 'S'
58
59      AND RL.ExtRetailPriceWTax < 0
60
61      AND R.SalesDate BETWEEN @DateFrom AND @DateTo
62
63      AND R.STATUSCODE = 'A'
64
65      AND R.Version != 'SYSVENT'
66
67  GROUP BY R.StoreNo, R.ReceiptId;
68

69  UPDATE R
70
71  SET
72
73      R.DiscTotal = R.DiscTotal + ISNULL(RT.DiscAmountLine, 0) * (1 + (R.TaxPercent
74
75      ↪ / 100)),
76
77      R.SubTotal = R.SubTotal + ISNULL(RT.DiscAmountLine, 0) * (1 + (R.TaxPercent /
78
79      ↪ 100)),
80
81      R.SubTotalWTax = R.SubTotalWTax + ISNULL(RT.DiscAmountLine, 0),
82
83      R.DiscPercent = ABS(((R.PayTotal / (R.SubTotalWTax +
84
85      ↪ ISNULL(RT.DiscAmountLine, 0)))) - 1) * 100)
86
87  FROM #ttRR R
88
89  INNER JOIN #RECEIPT_RETURN RT ON RT.ReceiptId = R.ReceiptId AND RT.StoreNo =
90
91      ↪ R.StoreNo;
92

93
94  END
95
96
97  -- Ventas con detalle y márgenes
98
99  SELECT DISTINCT
100
101      RL.StoreNo AS Tienda,
```

```
69      CONVERT(char(8), RL.SalesDate, 112) AS Fecha,
70      RC.ReceiptId,
71      RL.SKU,
72      CASE WHEN RL.SalesCode = 'S' AND SKU = 1454037 THEN
73          SUM(ISNULL(RL.ExtRetailPrice, 0) * (1 - (ISNULL(RC.DiscPercent, 0) /
74          ↪ 100))) END AS ExtRetailPriceWGlobalDisc,
75      CASE WHEN RL.SalesCode = 'S' AND SKU = 1454037 THEN
76          SUM(ISNULL(RL.ExtRetailPrice, 0) * (1 - (ISNULL(RC.DiscPercent, 0) /
77          ↪ 100))) END AS ExtRetailPriceWGlobalDiscPuntos,
78      CASE WHEN RL.SalesCode = 'R' THEN
79          SUM(ISNULL(RL.ExtRetailPrice, 0) * (1 - (ISNULL(RC.DiscPercent, 0) /
80          ↪ 100))) END AS ExtRetailPriceWGlobalDiscCupon,
81      CASE WHEN RL.SalesCode = 'S' AND SKU NOT IN (1454037) THEN
82          SUM(ISNULL(RL.QTY, 0)) END AS CantidadUnidades,
83      CASE WHEN RL.SalesCode = 'S' AND SKU = 1454037 THEN
84          SUM(ISNULL(RL.QTY, 0)) END AS CantidadPuntos,
85      CASE WHEN RL.SalesCode = 'R' THEN
86          SUM(ISNULL(RL.QTY, 0)) END AS CantidadUnidadesCupon,
87      CASE WHEN RL.SalesCode = 'S' AND SKU = 1454037 THEN
88          SUM(ISNULL(RL.ExtRetailPrice, 0) * (1 - (ISNULL(RC.DiscPercent, 0) /
89          ↪ 100))) - SUM(ExtCost) END AS ExtMargin
90      INTO #ttVentasDetalle
91      FROM #ttRR RC (NOLOCK)
92      INNER JOIN RECEIPT_LINE RL (NOLOCK) ON RL.ReceiptId = RC.ReceiptId AND RC.StoreNo
93      ↪ = RL.StoreNo
94      WHERE RC.StatusCode = 'A'
95      AND RC.Version != 'SYSVENT'
96      AND RC.SalesCode = 'S'
97      AND RC.SalesDate BETWEEN @DateFrom AND @DateTo
98      AND (RC.StoreNo = @StoreNo OR @StoreNo = 9999)
99      GROUP BY RL.StoreNo, CONVERT(char(8), RL.SalesDate, 112), RC.ReceiptId,
100      ↪ RL.SalesCode, RL.SKU
101
102      UNION ALL
```

```

98    SELECT DISTINCT
99        RL.StoreNo,
100       CONVERT(char(8), RL.SalesDate, 112),
101       RC.ReceiptId,
102       RL.SKU,
103       SUM(RL.ExtRetailPrice * (1 - ISNULL(RC.DiscPercent, 0) / 100)),
104       0,
105       0,
106       SUM(ISNULL(RL.QTY, 0)),
107       0,
108       0,
109       SUM(ISNULL(RL.ExtRetailPrice, 0) * (1 - ISNULL(RC.DiscPercent, 0) / 100)) -
110      ↳  SUM(ExtCost)
111
112     FROM RECEIPT RC (NOLOCK)
113     INNER JOIN RECEIPT_LINE RL (NOLOCK) ON RL.ReceiptId = RC.ReceiptId AND RC.StoreNo
114      ↳ = RL.StoreNo
115     WHERE RC.StatusCode = 'A'
116       AND RC.Version != 'SYSEVENT'
117       AND RC.SalesCode = 'R'
118       AND RC.SalesDate BETWEEN @DateFrom AND @DateTo
119       AND (RC.StoreNo = @StoreNo OR @StoreNo = 9999)
120
121     GROUP BY RL.StoreNo, CONVERT(char(8), RL.SalesDate, 112), RC.ReceiptId,
122      ↳ RL.SalesCode, RL.SKU;
123
124
125     -- Agregación de ventas
126
127     SELECT
128        tienda,
129        Fecha,
130        ReceiptId,
131        SUM(ISNULL(ExtRetailPriceWGlobalDisc, 0)) AS ExtRetailPriceWGlobalDisc,
132        SUM(ISNULL(ExtRetailPriceWGlobalDiscPuntos, 0)) AS
133        ↳ ExtRetailPriceWGlobalDiscPuntos,
134        SUM(ISNULL(ExtRetailPriceWGlobalDiscCupon, 0)) AS
135        ↳ ExtRetailPriceWGlobalDiscCupon,
136        SUM(ISNULL(CantidadUnidades, 0)) AS CantidadUnidades,

```

```
128     SUM(ISNULL(CantidadPuntos, 0)) AS CantidadPuntos,
129     SUM(ISNULL(CantidadUnidadesCupon, 0)) AS CantidadUnidadesCupon,
130     SUM(ISNULL(ExtMargin, 0)) AS ExtMargin
131   INTO #RL
132   FROM #ttVentasDetalle
133   GROUP BY tienda, Fecha, ReceiptId;
134
135   -- Integración final con información del recibo y cliente
136   SELECT
137     receipt.ReceiptId,
138     receipt.StoreNo,
139     StoreCode,
140     StoreName,
141     receipt.ReceiptNo,
142     receipt.RegisterId,
143     receipt.CustomerNo,
144     receipt.SalesDate,
145     receipt.StatusCode,
146     document_status.StatusName,
147     receipt.SalesCode,
148     sales_code.SalesDescription,
149     receipt.RegisterSN,
150     receipt.DiscTotal,
151     receipt.DiscTotalWTax,
152     receipt.DiscPercent,
153     receipt.FeeTotal,
154     receipt.ManualReceipt,
155     receipt.FeeTaxAmount,
156     receipt.FeeTotalWTax,
157     receipt.TaxTotal,
158     receipt.TaxPercent,
159     receipt.SubTotal,
160     receipt.SubTotalWTax,
161     receipt.ShippingTotal,
162     receipt.PayTotal,
```

```
163     receipt.LineCount,  
164     receipt.CurrencyId,  
165     CurrencyName,  
166     receipt.flag1,  
167     receipt_flag1.Flag1Description,  
168     receipt.DocNumber,  
169     ISNULL(FirstName, '') AS FirstName,  
170     ISNULL(LastName, '') AS LastName,  
171     ISNULL(MiddleName, '') AS MiddleName,  
172     ISNULL(MaidenName, '') AS MaidenName,  
173     ISNULL(CompanyName, '') AS CompanyName,  
174     ISNULL(Customer.Info1, '') AS Info1,  
175     ISNULL(Customer.LicenseNumber, '') AS LicenseNumber,  
176     receipt.ExchangeRate,  
177     receipt.TakeBase,  
178     receipt.TakeExchange,  
179     #ttStore.StoreInfo1,  
180     #ttStore.StoreInfo2,  
181     receipt.Cashier,  
182     receipt.ACCT1Date,  
183     SUM(CantidadUnidades) AS CantidadUnidades,  
184     SUM(CantidadPuntos) AS CantidadPuntos,  
185     SUM(CantidadUnidadesCupon) AS CantidadUnidadesCupon,  
186     ref.ReceiptNo AS ReferenceReceiptNo,  
187     ref.StoreNo AS ReferenceStoreNo,  
188     ref.DocNumber AS ReferenceDocNumber,  
189     ref.RegisterSN AS ReferenceRegisterSN,  
190     ref.SalesDate AS ReferenceSalesDate,  
191     ref.RegisterId AS ReferenceRegisterId,  
192     ref.SalesCode AS ReferenceSalesCode,  
193     ref.Flag1 AS ReferenceFlag1,  
194     ref_flag1.Flag1Description AS ReferenceFlag1Description,  
195     ref.TakeBase AS ReferenceTakeBase,  
196     receipt.SourceStore,  
197     receipt.Notes
```

```
198     INTO #ttFinalResult
199
200     FROM receipt (NOLOCK)
201
202     LEFT JOIN #RL ON receipt.StoreNo = #RL.Tienda AND receipt.ReceiptId =
203         ↳ #RL.ReceiptId
204
205     INNER JOIN sales_code ON receipt.SalesCode = sales_code.SalesCode
206
207     INNER JOIN document_status ON receipt.StatusCode = document_status.StatusCode
208
209     INNER JOIN currency ON receipt.CurrencyId = currency.CurrencyId
210
211     INNER JOIN #ttStore ON receipt.StoreNo = #ttStore.StoreNo
212
213     LEFT JOIN Customer ON Customer.CustomerNo = receipt.CustomerNo
214
215     LEFT JOIN receipt_flag1 ON receipt.Flag1 = receipt_flag1.Flag1
216
217     LEFT JOIN receipt ref ON receipt.ReferenceId = CAST(ref.StoreNo AS varchar) + '@'
218         ↳ + CAST(ref.ReceiptId AS varchar(36)) AND receipt.ReferenceId IS NOT NULL
219
220     LEFT JOIN receipt_flag1 ref_flag1 ON ref.Flag1 = ref_flag1.Flag1
221
222     WHERE receipt.SalesDate BETWEEN @DateFrom AND @DateTo
223
224         AND receipt.StatusCode IN ('A', 'C')
225
226         AND receipt.SalesCode IN ('S', 'R')
227
228     GROUP BY
229
230         receipt.ReceiptId,
231
232         receipt.StoreNo,
233
234         StoreCode,
235
236         StoreName,
237
238         receipt.ReceiptNo,
239
240         receipt.RegisterId,
241
242         receipt.CustomerNo,
243
244         receipt.SalesDate,
245
246         receipt.StatusCode,
247
248         document_status.StatusName,
249
250         receipt.SalesCode,
251
252         sales_code.SalesDescription,
253
254         receipt.RegisterSN,
255
256         receipt.DiscTotal,
257
258         receipt.DiscTotalWTax,
259
260         receipt.DiscPercent,
261
262         receipt.FeeTotal,
263
264         receipt.ManualReceipt,
```

```
231     receipt.FeeTaxAmount,  
232     receipt.FeeTotalWTax,  
233     receipt.TaxTotal,  
234     receipt.TaxPercent,  
235     receipt.SubTotal,  
236     receipt.SubTotalWTax,  
237     receipt.ShippingTotal,  
238     receipt.PayTotal,  
239     receipt.LineCount,  
240     receipt.CurrencyId,  
241     CurrencyName,  
242     receipt.flag1,  
243     receipt_flag1.Flag1Description,  
244     receipt.DocNumber,  
245     ISNULL(FirstName, ""),  
246     ISNULL(LastName, ""),  
247     ISNULL(MiddleName, ""),  
248     ISNULL(CompanyName, ""),  
249     ISNULL(Customer.Info1, ""),  
250     ISNULL(Customer.LicenseNumber, ""),  
251     receipt.ExchangeRate,  
252     receipt.TakeBase,  
253     receipt.TakeExchange,  
254     #ttStore.StoreInfo1,  
255     #ttStore.StoreInfo2,  
256     receipt.Cashier,  
257     receipt.ACCT1Date,  
258     ref.ReceiptNo,  
259     ref.StoreNo,  
260     ref.DocNumber,  
261     ref.RegisterSN,  
262     ref.SalesDate,  
263     ref.RegisterId,  
264     ref.SalesCode,  
265     ref.Flag1,
```

```
266     ref_flag1.Flag1Description,  
267     ref.TakeBase,  
268     receipt.SourceStore,  
269     receipt.Notes,  
270     receipt.RegisterSN,  
271     Customer.MaidenName;  
272  
273     -- Reinicio de valores si el estado no es A  
274     UPDATE #ttFinalResult  
275     SET  
276         DiscTotal = 0,  
277         DiscTotalWTax = 0,  
278         DiscPercent = 0,  
279         FeeTotal = 0,  
280         TaxTotal = 0,  
281         TaxPercent = 0,  
282         SubTotal = 0,  
283         SubTotalWTax = 0,  
284         ShippingTotal = 0,  
285         PayTotal = 0,  
286         FeeTaxAmount = 0  
287     WHERE StatusCode != 'A';  
288  
289     SELECT * FROM #ttFinalResult  
290     ORDER BY RegisterId, SalesDate, ReceiptNo;
```

4.6. Extracción y transformación de datos de los documentos de la Base de datos de SAP.

En esta etapa se establece la conexión con el servidor de SAP y se configuran las fechas de inicio y fin. Posteriormente, se ejecuta un procedimiento almacenado utilizando como parámetros dichas fechas. Finalmente, los resultados son recuperados y cargados en un dataframe para su procesamiento.

Código 24. Transformación de datos de los documentos de la Base de datos del sistema SAP.

```
1 def consultaSAP_BD():
2     server = IP_SERVER_SAP
3     database = DATABASE_SAP
4     username = USERNAME_DATABASE_SAP
5     password = PASSWORD_DATABASE_SAP
6
7     # cadena de conexión
8     conn = pyodbc.connect(
9         f"DRIVER={{SQL Server Native Client
10             \r 11.0}};SERVER={server};DATABASE={database};UID={username};PWD={password}"
11     )
12
13     cursor = conn.cursor()
14
15     # setear fecha ini/fin un dia anterior
16     fini = datetime.now() - timedelta(days=7)
17     fini = fini.strftime("%Y%m%d")
18
19     fFin = datetime.now() - timedelta(days=1)
20     fFin = fFin.strftime("%Y%m%d")
21
22     query = "EXEC sp_appConsultaSAP @fini = '{0}', @fFin = '{1}' ".format(fini,fFin)
```

```
22     # mostrar dataframe usando lib pandas
23     dfSAP = pd.read_sql_query(query, conn)
24
25     return dfSAP
26
27     conn.close()
```

El siguiente SP consulta registros de la tabla ADMSAP.RMSFILE en un rango de fechas: Sp_appConsultaSAP.

Código 25. Procedimiento almacenado para consultar datos de SAP (RMSFILE)

```
1  CREATE PROCEDURE sp_appConsultaSAP
2
3      @fini date,
4      @ffin date
5
6  AS
7
8  BEGIN
9
10     SET NOCOUNT ON;
11
12     -- Código comentado que sugiere procesamiento y transformación de datos
13     -- de la tabla RMSFILE antes de la consulta
14
15     --SELECT RMSFDOC, rmsffec,
16     --RTRIM(LEFT(RMSFREF, 8) + REPLICATE('0', 8 - LEN(RIGHT(RMSFREF, 7))) +
17     --    RIGHT(RMSFREF, 7)),
18     --RMSFNOM, RMSFMONT, RMSFMONE, RMSFTIPC
19     --FROM [AS400_PRD].[S7824B51].[ADMSAP].[RMSFILE]
20
21     SELECT *
22     FROM [AS400_PRD].[S7824B51].[ADMSAP].[RMSFILE]
23     WHERE rmsffec BETWEEN @fini AND @ffin;
24
25     --DROP TABLE #TempRMSFILE
26
27 END
```

Explicación: Este procedimiento consulta todos los registros de la tabla RMSFILE dentro de un rango de fechas proporcionado por el usuario. El código comentado sugiere que en versiones anteriores se realizaba un procesamiento adicional del campo RMSFREF, probablemente para formatearlo con ceros a la izquierda. Actualmente, la consulta se realiza directamente sin transformación.

4.7. Sistematización e integración de la información de los tres sistemas.

En esta etapa se realiza el cruce de información de los 3 sistemas, tanto de Acepta, RMS y SAP. Donde buscamos información faltante entre los 3 sistemas ya mencionados.

Código 26. Cruce de información de los tres sistemas.

```
1 def Cruce():
2
3     df1 = consultaSAP_BD()
4
5     df2 = consultaRMS_BD()
6
7     df3 = descargaAcepta()
8
9
10    #-----SAP - RMS-----
11
12    # busca los nro_cpe de RMS que no estan en RMSFREF de SAP
13    df_RmsDocNotInSAP = df2[~df2.nro_cpe.isin(df1.RMSFREF) & df2.nro_cpe.notna() &
14        (df2.nro_cpe != "")]
15
16
17    # realizar la fusión en función del código (nro_cpe/rms y RMSFREF/sap)
18    merged_df = pd.merge(df1, df2, left_on="RMSFREF", right_on="nro_cpe",
19        suffixes=("_df1", "_df2"))
20
21    resultadoFinalSAP = df_RmsDocNotInSAP
22
23
24    #-----Acepta - RMS -----
25
26    #eliminar documentos que empiecen con FW y FX
27    df4 = df3[~(df3['nro_cpe'].str.startswith('FX') | df3['nro_cpe'].str.startswith('FW') |
28        | df3['nro_cpe'].str.startswith('BX') | df3['nro_cpe'].str.startswith('BG'))]
29
30    df_rechazado = df4[df4["estado_acepta"] != "ACEPTADO"]
31
32    df_AcepDocNotInRMS = df4[~df4.nro_cpe_.isin(df2.nro_cpe) & df4.nro_cpe_.notna() &
33        (df4.nro_cpe_ != "")]
34
35    resultadoFinalAcepta = pd.concat([df_rechazado,df_AcepDocNotInRMS])
36
37
38    #-----RMS - ACEPTA -----
39
```

```
22     resultadoFinalRMS = df2[~df2.nro_cpe.isin(df3.nro_cpe_) & df2.nro_cpe.notna() &
23     ↪ (df2.nro_cpe != '')]
24
25     #-----SAP - ACEPTA-----
26     #Documentos que estan en sap pero no en acepta
27     resultadoFinalSapAcep = df1[~df1.RMSFREF.isin(df3.nro_cpe_) & df1.RMSFREF.notna()
28     ↪ & (df1.RMSFREF != '')]
```

4.8. Generación de reportes y/o alerta.

Finalmente, se genera el html con la estructura que se debe enviar en el reporte respecto a la data faltante en los 3 sistemas.

Código 27. Generación de reportes.

```
1 #-----ENVIAR CORREO (ALERTA)-----  
2 html = """\n3 <html>\n4 <head></head>\n5 <body>\n6     <h2 style='text-align: left; color : #5499C7;'>TRADING FASHION LINE S.A.</h2>\n7     <p>Fecha : {0}</p>\n8     <p> Rango de Cruce: 7 Dias</p>\n9     <h3>ACEPTA - RMS</h3>\n10    <p>Documentos que están en ACEPTA y no están en RMS y/o estado distinto.</p>\n11    {1}\n12    <br>\n13    <br>\n14    <h3>RMS - ACEPTA </h3>\n15    <p>Documentos que están en RMS y no están en ACEPTA y/o estado distinto.</p>\n16    {2}\n17    <br>\n18    <h3>SAP - RMS </h3>\n19    <p>Documentos que están en RMS y no están en SAP y/o estado distinto.</p>\n20    {3}\n21    <br>\n22    <h3>SAP - ACEPTA </h3>\n23    <p>Documentos que están en SAP y no están en ACEPTA y/o estado distinto.</p>\n24    {4}\n25    <br>\n26    <p style='text-align: center; color: #999999;'>Nota: Este correo ha sido generado  
    ↵ automáticamente, no responder a este correo</p>
```

```
27     <h2 style='text-align: center; color: #999999;'>© 2023 TI RETAIL TFL. All rights
28     ↵ reserved.</h2>
29 </body>
30 </html>
31 """ .format((datetime.now()).strftime("%d-%m-%Y %H:%M") ,resultadoFinalAcepta.to_html(i,
32     ↵ index=False) ,resultadoFinalRMS.to_html(index=False) ,
33     ↵ resultadoFinalSAP.to_html(index=False) ,resultadoFinalSapAcep.to_html(index=False))
```

Código 28. Envío de la alerta

```
1
2 part1 = MIMEText(html, 'html')
3 message= part1
4 message['Subject'] = Header("Cruce RMS - Acepta -SAP",'utf-8')
5
6 server=smtplib.SMTP('191.1.64.16',25)
7 server.starttls()
8 server.login(usuario_alertas,contrasenia_alertas)
9 destinatarios = destinatarios_alerta
10
11 server.sendmail(usuario_alertas,destinatarios,message.as_string())
12
13 server.quit()
```

4.9. Despliegue en Amazon Web Services (AWS).

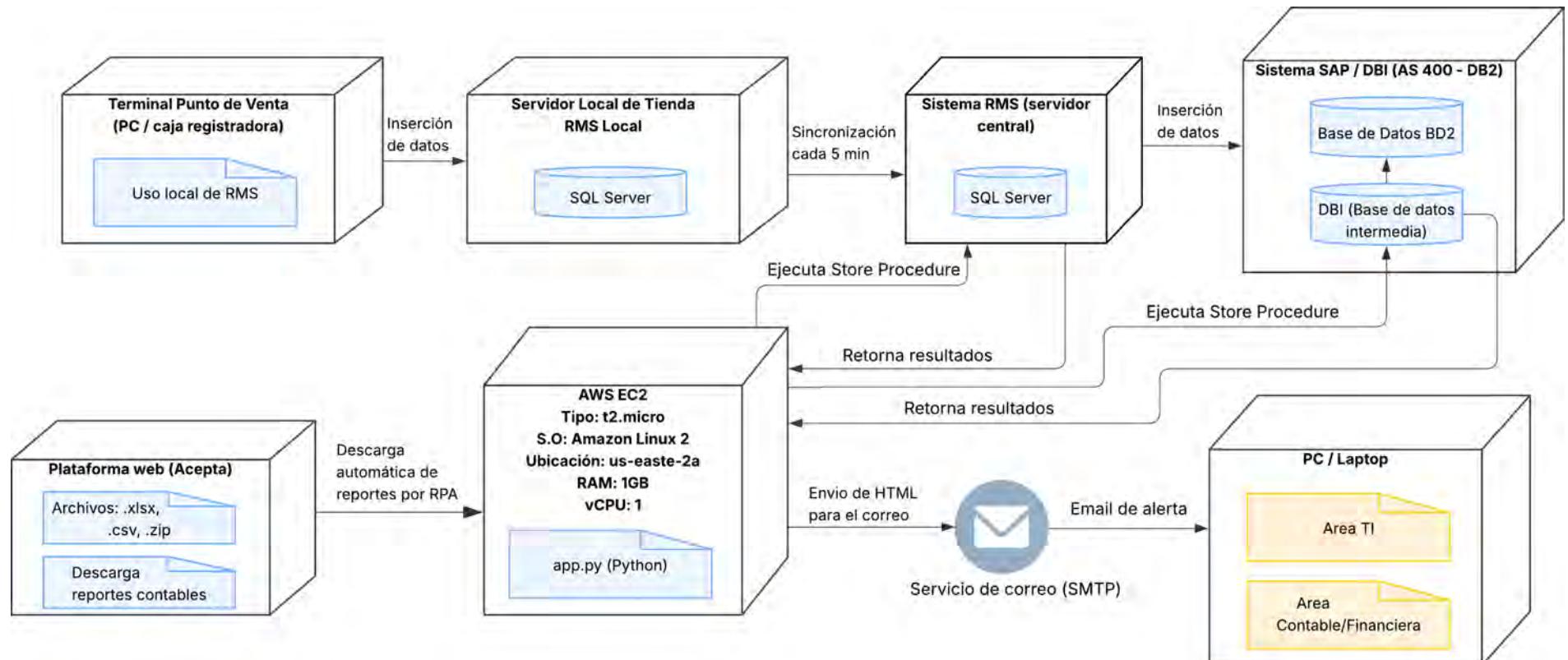
4.9.1. Diagrama de despliegue

El diagrama de despliegue presentado ilustra la arquitectura propuesta para la integración y automatización de los procesos contables y financieros en la empresa Topitop, utilizando servicios en la nube de Amazon Web Services (AWS), RPA y procedimientos almacenados. El flujo inicia con la extracción de datos desde los terminales de punto de venta (PC/Caja Registradora) y la plataforma web externa (Acepta), cuyos reportes contables son descargados y procesados automáticamente mediante un script en Python desplegado en una instancia EC2 (Amazon Linux 2).

La solución contempla la sincronización de datos entre el servidor local de tienda (RMS Local) y el sistema RMS central, ambos basados en SQL Server, a través de la ejecución programada de procedimientos almacenados. Posteriormente, los datos son transferidos hacia el sistema SAP/DBI, donde se almacenan en una base de datos DB2 mediante la capa intermedia DBI.

Adicionalmente, el sistema automatizado genera alertas y notificaciones por correo electrónico utilizando un servicio SMTP, enviando reportes de cruce contable tanto al área de TI como al área contable. De esta manera, se garantiza la integración homogénea de la información, el procesamiento centralizado y la automatización de las alertas para la gestión eficiente de los datos contables y financieros de la organización.

Figura 16
Diagrama de despliegue



Nota. Elaboración propia.

4.9.2. Crear una instancia EC2

1. Ve a AWS EC2 y haz clic en "Launch Instance".

Figura 17
E2C Instancias

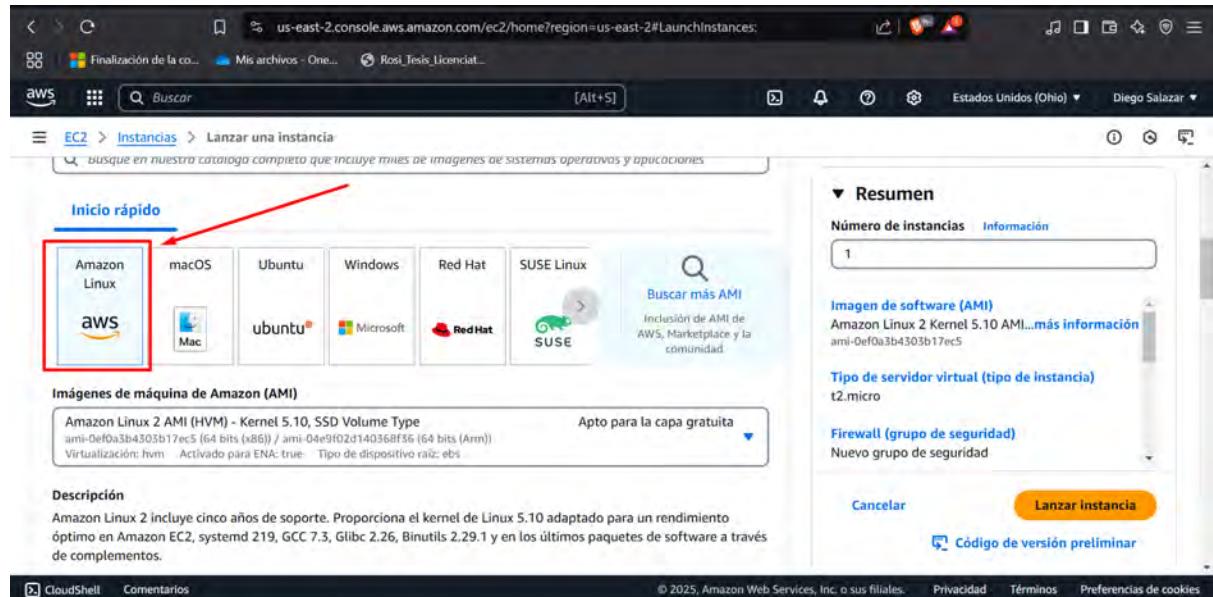


Nota. Amazon Web Service.

2. Selecciona una imagen de máquina Ubuntu 22.04 o Amazon Linux 2. Debido a que Amazon Linux 2 incluye cinco años de soporte. Proporciona el kernel de Linux 5.10 adaptado para un rendimiento óptimo en Amazon EC2, systemd 219, GCC 7.3, Glibc 2.26, Binutils 2.29.1 y en los últimos paquetes de software a través de complementos. Amazon Linux 2 Kernel 5.10 AMI 2.0.20250220.0 x86_64 HVM gp2.

Figura 18

Seleccionar instancia de maquina virtual

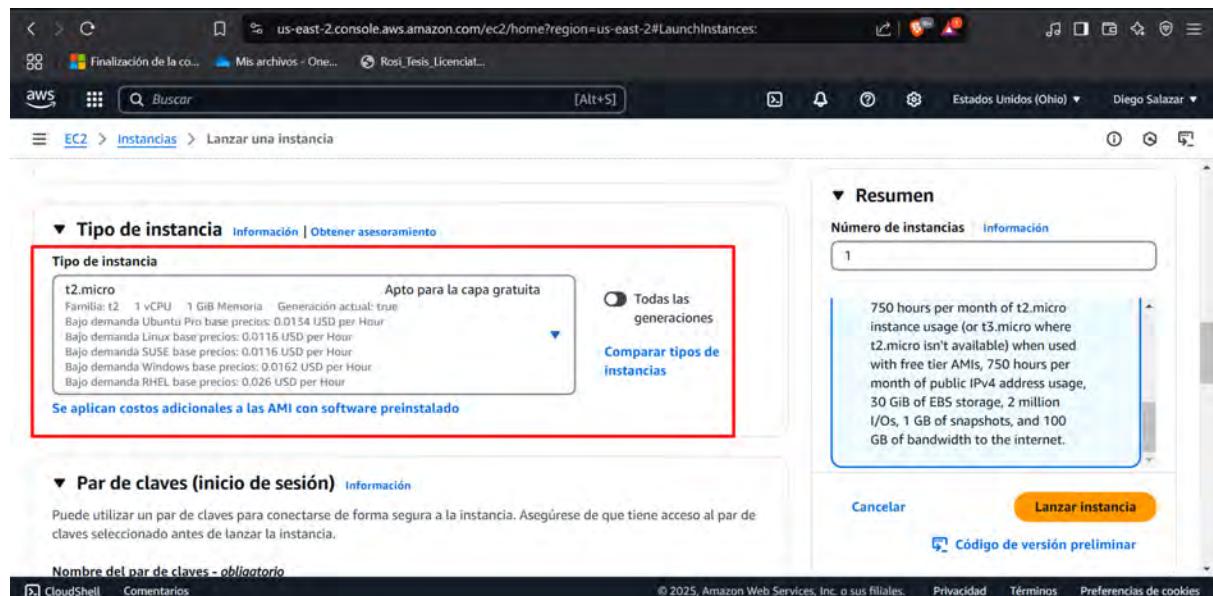


Nota. Amazon Web Service.

3. Elige un tipo de instancia (ejemplo: t3.medium para más RAM si usas muchos procesos).

Figura 19

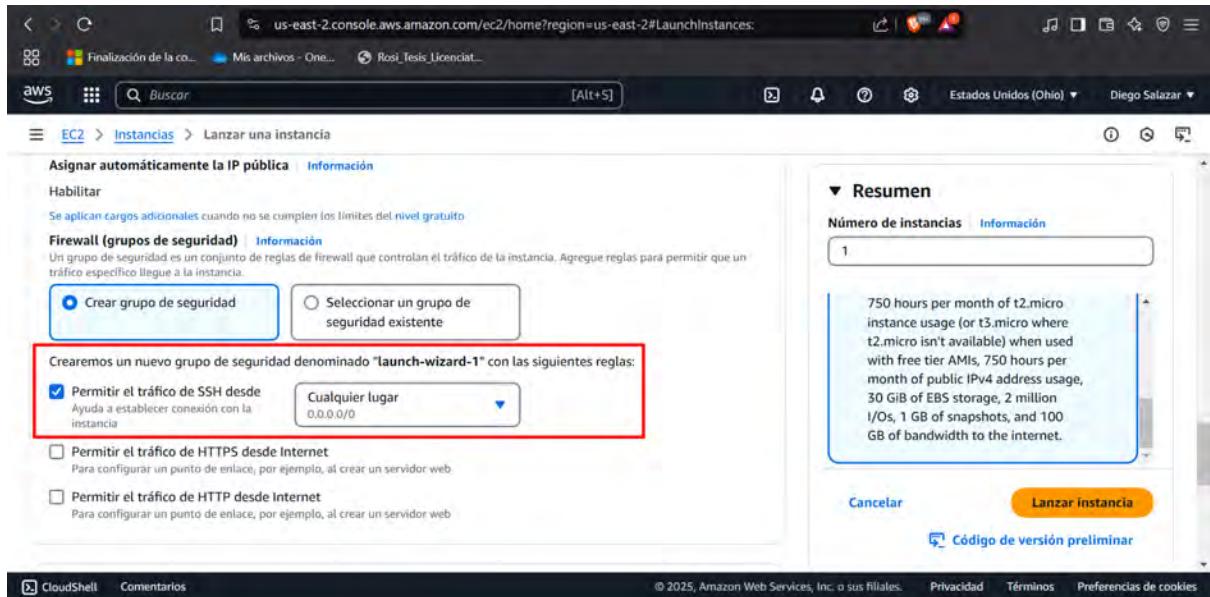
Seleccionar tipo de instancia



Nota. Amazon Web Service.

4. Habilita acceso SSH para conectarte a la instancia.

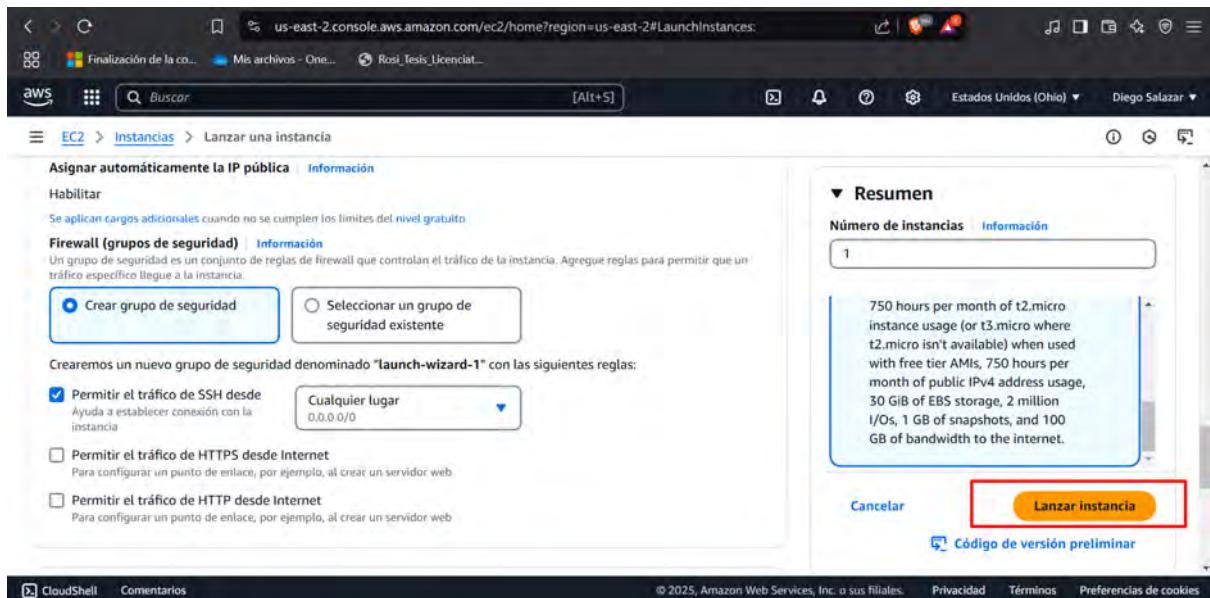
Figura 20
Habilitar acceso SSH para la instancia



Nota. Amazon Web Service.

5. Hacer clic en Launch.

Figura 21
Crear instancia



Nota. Amazon Web Service.

4.9.3. Conectarse a la Instancia

Conectarse desde Windows con PowerShell o CMD

Paso 1: Abrir CMD (Win + X → "CMD").

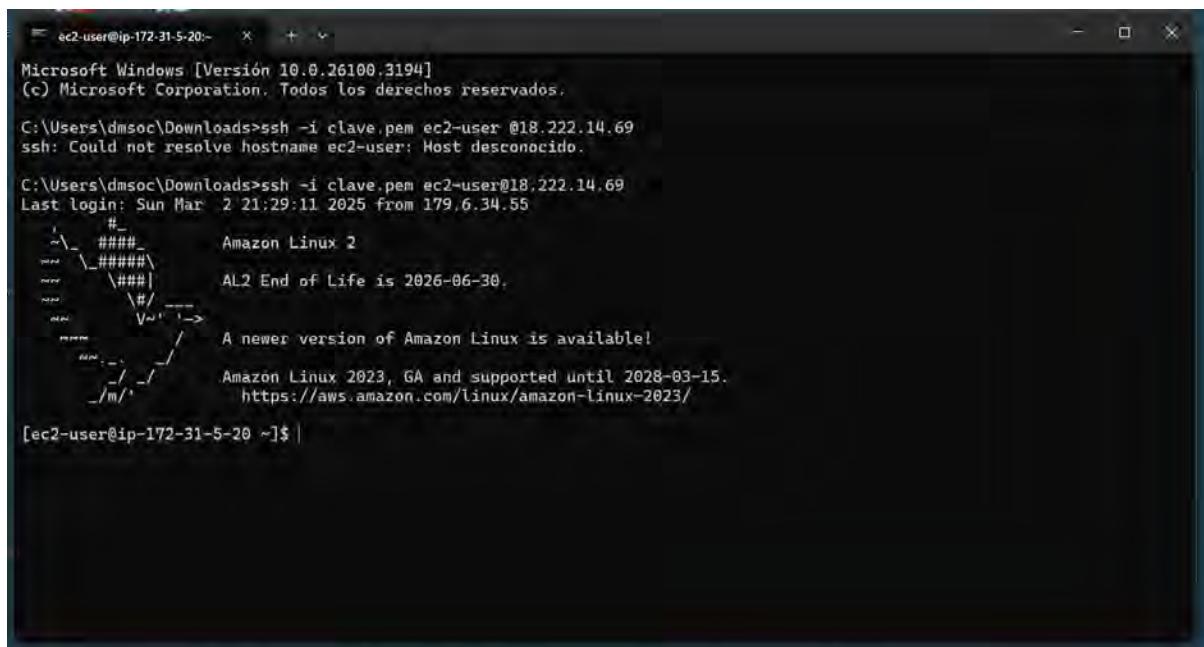
Paso 2: Navegar hasta la carpeta donde está la clave (.pem).

Paso 3: Conectarse con SSH: Como se esta usando Amazon Linux 2 , el usuario es "ec2-user" y la ip publica es 18.222.14.69, lo colocamos en la terminal de esta manera:

```
ssh -i clave.pem ec2-user@18.222.14.69
```

Nos debe aparecer una imagen o datos similares a esta, esto significa que nos conectamos a la instancia de aws mediandte ssh.

Figura 22
Conexión a la maquina virtual



The screenshot shows a Windows Command Prompt window with the following text:

```
ec2-user@ip-172-31-5-20:~ + - ×
Microsoft Windows [Versión 10.0.26100.3194]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user @18.222.14.69
ssh: Could not resolve hostname ec2-user: Host desconocido.

C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user@18.222.14.69
Last login: Sun Mar  2 21:29:11 2025 from 179.6.34.55
#_###_ Amazon Linux 2
##_###_ AL2 End of Life is 2026-06-30,
##_###_ #/--->
##_###_ V~`-->
##_###_ / A newer version of Amazon Linux is available!
##_###_ / Amazon Linux 2023, GA and supported until 2028-03-15.
##_###_ / https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-5-20 ~]$ |
```

Nota. Elaboración propia.

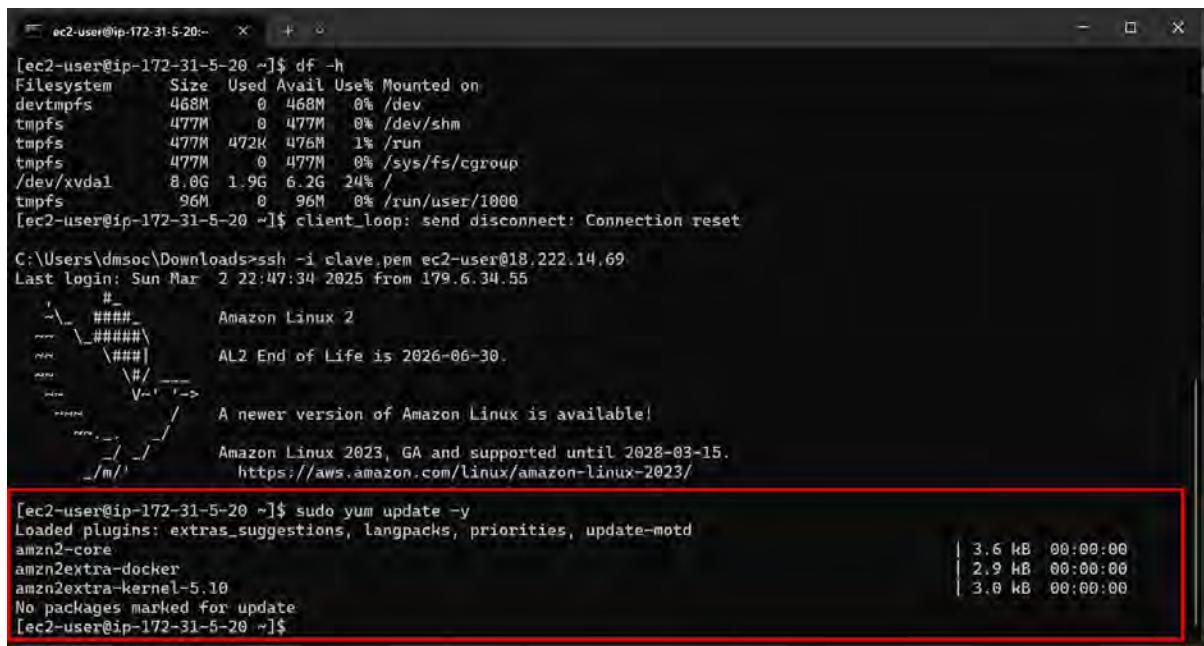
4.9.4. Preparar el servidor

4.9.4.1. Instalación de Python

- Instalación de Python y PIP para Amazon Linux.

```
sudo yum update -y
```

Figura 23
Actualización de paquetes



```
[ec2-user@ip-172-31-5-20 ~]$ df -h
Filesystem      Size  Used  Avail Use% Mounted on
devtmpfs        468M   0  468M  0% /dev
tmpfs          477M   0  477M  0% /dev/shm
tmpfs          477M  472K  476M  1% /run
tmpfs          477M   0  477M  0% /sys/fs/cgroup
/dev/xvda1      8.0G  1.9G  6.2G  24% /
tmpfs          96M   0   96M  0% /run/user/1000
[ec2-user@ip-172-31-5-20 ~]$ client_loop: send disconnect: Connection reset

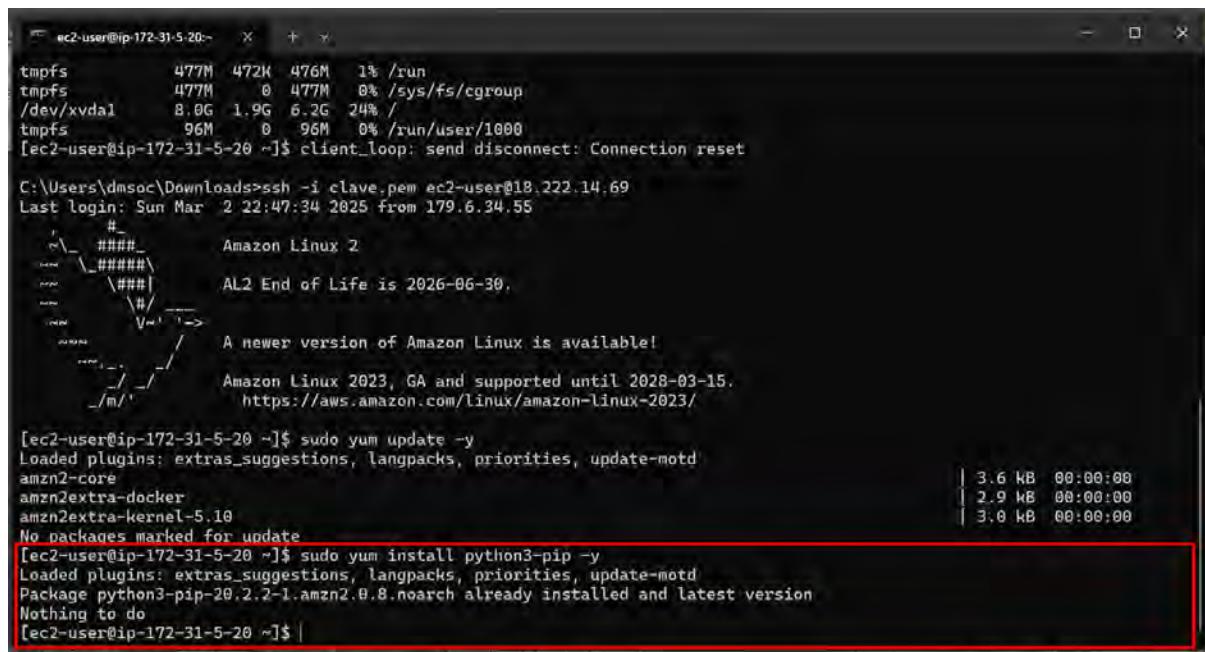
C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user@18.222.14.69
Last login: Sun Mar  2 22:47:34 2025 from 179.6.34.55
#_ _ _ _ _ Amazon Linux 2
--- \####\ AL2 End of Life is 2026-06-30.
\##/ \# / A newer version of Amazon Linux is available!
\##/ \# / Amazon Linux 2023, GA and supported until 2028-03-15.
\##/ \# / https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-5-20 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-kernel-5.10
No packages marked for update
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

- Se ha gestionado con éxito los paquetes a actualizar. En este caso nos indica que Python ya está instalado por defecto.

```
sudo yum install python3-pip -y
```

Figura 24
Instalación de Python por terminal



```
ec2-user@ip-172-31-5-20: ~ + %
tmpfs      477M  472K  476M  1% /run
tmpfs      477M  0  477M  0% /sys/fs/cgroup
/dev/xvda1  8.0G  1.9G  6.2G  24% /
tmpfs      96M  0  96M  0% /run/user/1000
[ec2-user@ip-172-31-5-20 ~]$ client_loop: send disconnect: Connection reset

C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user@18.222.14.69
Last login: Sun Mar  2 22:47:34 2025 from 179.6.34.55
#
#_ _ _ _ _ Amazon Linux 2
#_ _ _ _ _ AL2 End of Life is 2026-06-30.
#_ _ _ _ _ A newer version of Amazon Linux is available!
#_ _ _ _ _ Amazon Linux 2023, GA and supported until 2028-03-15.
#_ _ _ _ _ https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-5-20 ~]$ sudo yum update -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-kernel-5.10
No packages marked for update
[ec2-user@ip-172-31-5-20 ~]$ sudo yum install python3-pip -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Package python3-pip-20.2.2-1.amzn2.0.8.noarch already installed and latest version
Nothing to do.
[ec2-user@ip-172-31-5-20 ~]$ |
```

Nota. Elaboración propia.

4.9.4.2. Instalación Google Chrome y ChromeDriver

Se habilita módulos adicionales en Amazon. Hasta que se visualice que la instalación fue correcta.

```
sudo amazon-linux-extras enable epel
```

Figura 25
Habilitación de módulos adicionales en Amazon

```

[ec2-user@ip-172-31-5-20: ~]$ sudo amazon-linux-extras enable epel
 2 httpd_modules          available  [ =1.0  =stable ]
 3 memcached1.5           available  \
  [ =1.5.1  =1.5.16  =1.5.17 ]
 9 R3.4                   available  [ =3.4.3  =stable ]
10 rust1                  available  \
  [ =1.22.1  =1.26.0  =1.26.1  =1.27.2  =1.31.0  =1.38.0
  =stable ]
18 libreoffice            available  \
  [ =5.0.6.2_15  =5.3.6.1  =stable ]
19 gimp                   available  [ =2.8.22 ]
20 +docker=latest         enabled   \
  [ =17.12.1  =18.03.1  =18.06.1  =18.09.9  =stable ]
21 mate-desktop1.x        available  \
  [ =1.19.0  =1.20.0  =stable ]
22 GraphicsMagick1.3     available  \
  [ =1.3.29  =1.3.32  =1.3.34  =stable ]
24 epel=latest             enabled   [ =7.11  =stable ]
25 testing                available  [ =1.0  =stable ]
26 ecs                   available  [ =stable ]
27 +correto8              available  \
  [ =1.8.0_192  =1.8.0_202  =1.8.0_212  =1.8.0_222  =1.8.0_232
  =1.8.0_242  =stable ]
32 lustre2.10             available  \
  [ =2.10.5  =2.10.8  =stable ]
34 lynis                  available  [ =stable ]
36 BCC                   available  [ =0.x  =stable ]
37 mono                  available  [ =5.x  =stable ]
38 nginx1                available  [ =stable ]
40 mock                  available  [ =stable ]

```

Nota. Elaboración propia.

Figura 26
Habilitación de módulos concluida

```

43 livepatch              available  [ =stable ]
45 haproxy2               available  [ =stable ]
46 collectd               available  [ =stable ]
47 aws-nitro-enclaves-cli available  [ =stable ]
48 R4                     available  [ =stable ]
49 kernel-5.4              available  [ =stable ]
50 selinux-ng              available  [ =stable ]
52 tomcat9                available  [ =stable ]
53 unbound1.13             available  [ =stable ]
54 +mariadb10.5            available  [ =stable ]
55 +kernel-5.10=latest     enabled   [ =stable ]
56 redis6                 available  [ =stable ]
59 +postgresql13           available  [ =stable ]
60 mock2                  available  [ =stable ]
61 dnsmasq2.85             available  [ =stable ]
62 kernel-5.15              available  [ =stable ]
63 +postgresql14           available  [ =stable ]
64 firefox                 available  [ =stable ]
65 lustre                  available  [ =stable ]
67 awscli                  available  [ =stable ]
68 +php8.2                 available  [ =stable ]
69 dnsmasq                 available  [ =stable ]
70 unbound1.17              available  [ =stable ]
72 +collectd-python3        available  [ =stable ]
+ Note on end-of-support. Use 'info' subcommand.

Now you can install:
# yum clean metadata
# yum install epel-release
[ec2-user@ip-172-31-5-20 ~]$ 

```

Nota. Elaboración propia.

Instalación de Google

Paso 1: Descargar el instalador de Google Chrome desde el sitio oficial de Google:

```
wget https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm
```

Figura 27

Descarga de Google Chrome con wget desde el sitio oficial



A terminal window showing the command `wget https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm` being run. The output shows the progress of the download, including the URL, connection details, and the final saved file name `google-chrome-stable_current_x86_64.rpm`.

```
[ec2-user@ip-172-31-5-20 ~]$ wget https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm
--2025-03-05 02:22:28-- https://dl.google.com/linux/direct/google-chrome-stable_current_x86_64.rpm
Resolving dl.google.com (dl.google.com)... 142.250.191.206, 2607:f800:4009:81a::200e
Connecting to dl.google.com (dl.google.com)|142.250.191.206|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 117119185 (112M) [application/x-rpm]
Saving to: 'google-chrome-stable_current_x86_64.rpm'

100%[=====] 117,119,185 64.5MB/s in 1.7s

2025-03-05 02:22:30 (64.5 MB/s) - 'google-chrome-stable_current_x86_64.rpm' saved [117119185/117119185]
[ec2-user@ip-172-31-5-20 ~]$
```

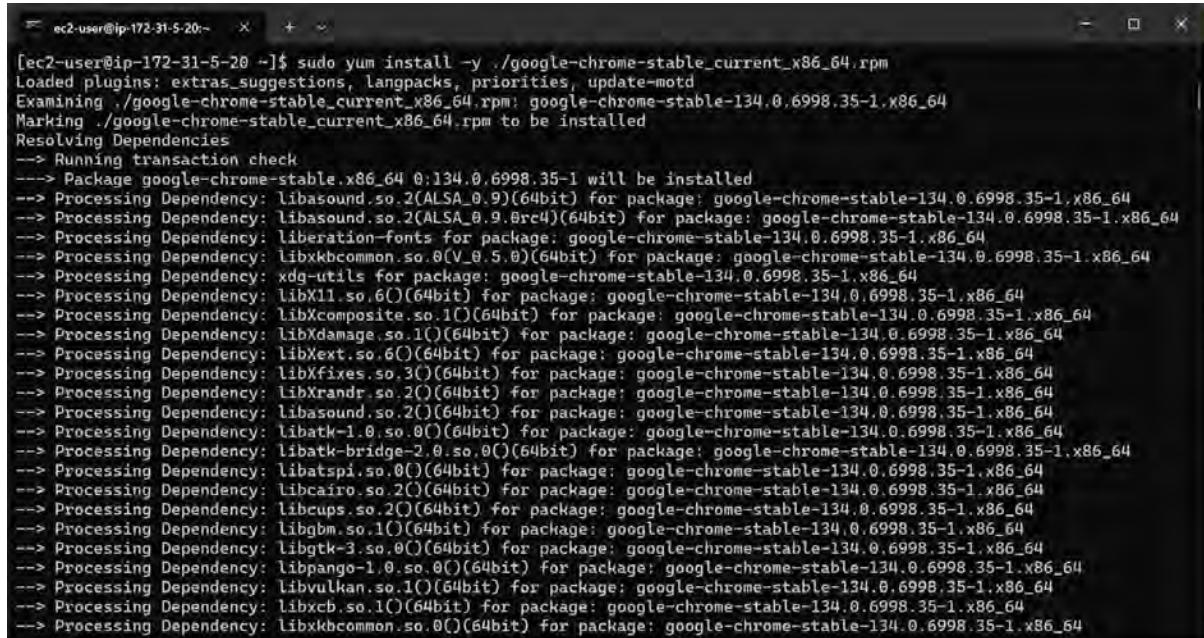
Nota. Elaboración propia.

Paso 2: Instalar el paquete descargado:

```
sudo yum install -y ./google-chrome-stable_current_x86_64.rpm
```

Figura 28

Instalación local de Google Chrome con yum

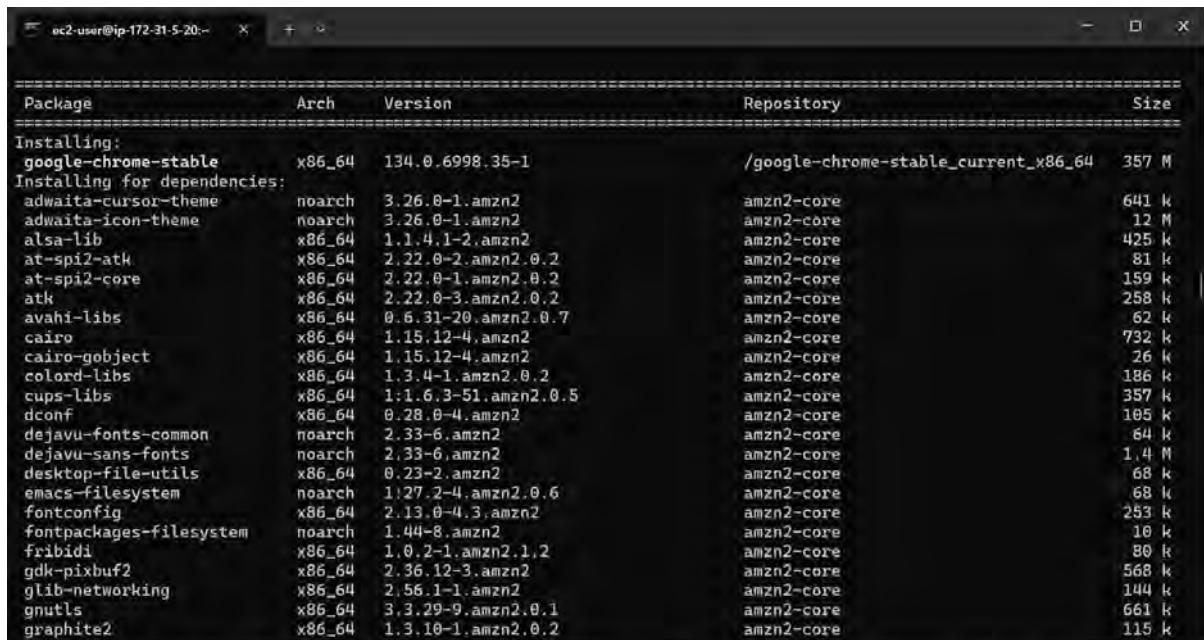


```
ec2-user@ip-172-31-5-20:~ % + ~
[ec2-user@ip-172-31-5-20 ~]$ sudo yum install -y ./google-chrome-stable_current_x86_64.rpm
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
Examining ./google-chrome-stable_current_x86_64.rpm: google-chrome-stable-134.0.6998.35-1.x86_64
Marking ./google-chrome-stable_current_x86_64.rpm to be installed
Resolving Dependencies
--> Running transaction check
--> Package google-chrome-stable.x86_64 0:134.0.6998.35-1 will be installed
--> Processing Dependency: libasound.so.2(ALSA_0.9)(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libasound.so.2(ALSA_0.9.0rc4)(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: liberation-fonts for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libxkbcommon.so.0(V_0.5.0)(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: xdg-utils for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libX11.so.6()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libXcomposite.so.1()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libXdamage.so.1()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libXext.so.6()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libXfixes.so.3()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libXrandr.so.2()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libasound.so.2()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libatk-1.0.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libatk-bridge-2.0.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libatspi.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libcairo.so.2()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libcups.so.2()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libgbm.so.1()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libgtk-3.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libpango-1.0.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libvulkan.so.1()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libxcb.so.1()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
--> Processing Dependency: libxkbcommon.so.0()(64bit) for package: google-chrome-stable-134.0.6998.35-1.x86_64
```

Nota. Elaboración propia.

Figura 29

Instalación local de Google Chrome en proceso

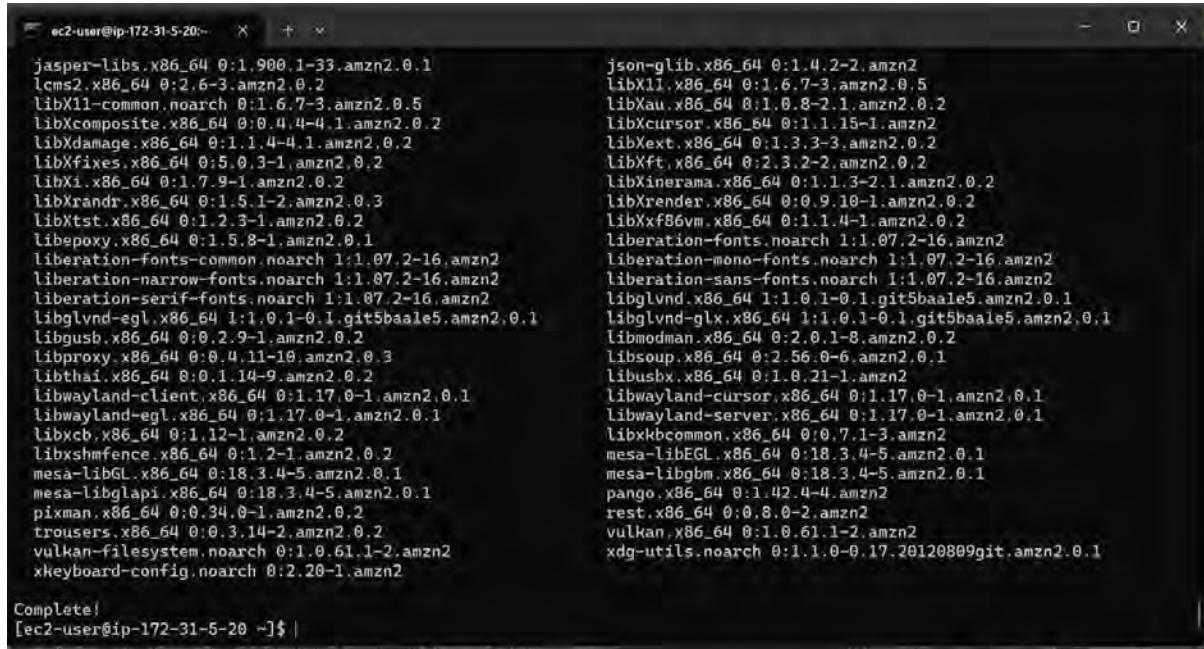


Package	Arch	Version	Repository	Size
Installing:				
google-chrome-stable	x86_64	134.0.6998.35-1	/google-chrome-stable_current_x86_64	357 M
Installing for dependencies:				
adwaita-cursor-theme	noarch	3.26.0-1.amzn2	amzn2-core	641 k
adwaita-icon-theme	noarch	3.26.0-1.amzn2	amzn2-core	12 M
alsa-lib	x86_64	1.1.4.1-2.amzn2	amzn2-core	425 k
at-spi2-atk	x86_64	2.22.0-2.amzn2.0.2	amzn2-core	81 k
at-spi2-core	x86_64	2.22.0-1.amzn2.0.2	amzn2-core	159 k
atk	x86_64	2.22.0-3.amzn2.0.2	amzn2-core	258 k
avahi-libs	x86_64	0.6.31-20.amzn2.0.7	amzn2-core	62 k
cairo	x86_64	1.15.12-4.amzn2	amzn2-core	732 k
cairo-gobject	x86_64	1.15.12-4.amzn2	amzn2-core	26 k
colord-libs	x86_64	1.3.4-1.amzn2.0.2	amzn2-core	186 k
cups-libs	x86_64	1:1.6.3-51.amzn2.0.5	amzn2-core	357 k
dconf	x86_64	0.28.0-4.amzn2	amzn2-core	105 k
dejavu-fonts-common	noarch	2.33-6.amzn2	amzn2-core	64 k
dejavu-sans-fonts	noarch	2.33-6.amzn2	amzn2-core	1.4 M
desktop-file-utils	x86_64	0.23-2.amzn2	amzn2-core	68 k
emacs-fs	noarch	1:27.2-4.amzn2.0.6	amzn2-core	68 k
fontconfig	x86_64	2.13.0-4.3.amzn2	amzn2-core	253 k
fontpackages-fs	noarch	1.44-8.amzn2	amzn2-core	10 k
fribidi	x86_64	1.0.2-1.amzn2.1.2	amzn2-core	80 k
gdk-pixbuf2	x86_64	2.36.12-3.amzn2	amzn2-core	568 k
glib-networking	x86_64	2.56.1-1.amzn2	amzn2-core	144 k
gnutls	x86_64	3.3.29-9.amzn2.0.1	amzn2-core	661 k
graphite2	x86_64	1.3.10-1.amzn2.0.2	amzn2-core	115 k

Nota. Elaboración propia.

Figura 30

Instalación local de Google Chrome concluida



```
ec2-user@ip-172-31-5-20: ~ % 
jasper-libs.x86_64 0:1.900.1-33.amzn2.0.1
lcms2.x86_64 0:2.6-3.amzn2.0.2
libX11-common.noarch 0:1.6.7-3.amzn2.0.5
libXcomposite.x86_64 0:0.4.4-4.1.amzn2.0.2
libXdamage.x86_64 0:1.1.4-4.1.amzn2.0.2
libXfixes.x86_64 0:5.0.3-1.amzn2.0.2
libXi.x86_64 0:1.7.9-1.amzn2.0.2
libXrandr.x86_64 0:1.5.1-2.amzn2.0.3
libXtst.x86_64 0:1.2.3-1.amzn2.0.2
libepoxy.x86_64 0:1.5.8-1.amzn2.0.1
liberation-fonts-common.noarch 1:1.07.2-16.amzn2
liberation-narrow-fonts.noarch 1:1.07.2-16.amzn2
liberation-serif-fonts.noarch 1:1.07.2-16.amzn2
libglvnd-egl.x86_64 1:1.0.1-0.1.git5baale5.amzn2.0.1
libgusb.x86_64 0:0.2.9-1.amzn2.0.2
libproxy.x86_64 0:0.4.11-10.amzn2.0.3
libthai.x86_64 0:0.1.14-9.amzn2.0.2
libwayland-client.x86_64 0:1.17.0-1.amzn2.0.1
libwayland-egl.x86_64 0:1.17.0-1.amzn2.0.1
libxcb.x86_64 0:1.12-1.amzn2.0.2
libxshmfence.x86_64 0:1.2-1.amzn2.0.2
mesa-libGL.x86_64 0:18.3.4-5.amzn2.0.1
mesa-libglapi.x86_64 0:18.3.4-5.amzn2.0.1
pixman.x86_64 0:0.34.0-1.amzn2.0.2
trousers.x86_64 0:0.3.14-2.amzn2.0.2
vulkan-filesystem.noarch 0:1.0.61.1-2.amzn2
xkeyboard-config.noarch 0:2.20-1.amzn2

json-glib.x86_64 0:1.4.2-2.amzn2
libX11.x86_64 0:1.6.7-3.amzn2.0.5
libXau.x86_64 0:1.0.8-2.1.amzn2.0.2
libXcursor.x86_64 0:1.1.15-1.amzn2
libXext.x86_64 0:1.3.3-3.amzn2.0.2
libXft.x86_64 0:2.3.2-2.amzn2.0.2
libXinerama.x86_64 0:1.1.3-2.1.amzn2.0.2
libXrender.x86_64 0:0.9.10-1.amzn2.0.2
libXxf86vm.x86_64 0:1.1.4-1.amzn2.0.2
liberation-fonts.noarch 1:1.07.2-16.amzn2
liberation-mono-fonts.noarch 1:1.07.2-16.amzn2
liberation-sans-fonts.noarch 1:1.07.2-16.amzn2
libglvnd.x86_64 1:1.0.1-0.1.git5baale5.amzn2.0.1
libglvnd-glx.x86_64 1:1.0.1-0.1.git5baale5.amzn2.0.1
libmodman.x86_64 0:2.0.1-8.amzn2.0.2
libsoup.x86_64 0:2.56.0-6.amzn2.0.1
libusbx.x86_64 0:1.0.21-1.amzn2
libwayland-cursor.x86_64 0:1.17.0-1.amzn2.0.1
libwayland-server.x86_64 0:1.17.0-1.amzn2.0.1
libxkbcommon.x86_64 0:0.7.1-3.amzn2
mesa-libEGL.x86_64 0:18.3.4-5.amzn2.0.1
mesa-libgbm.x86_64 0:18.3.4-5.amzn2.0.1
pango.x86_64 0:1.42.4-4.amzn2
rest.x86_64 0:0.8.0-2.amzn2
vulkan.x86_64 0:1.0.61.1-2.amzn2
xdg-utils.noarch 0:1.1.0-0.17.20120809git.amzn2.0.1

Complete!
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

Instalación de Google ChromeDriver para la librería Selenium

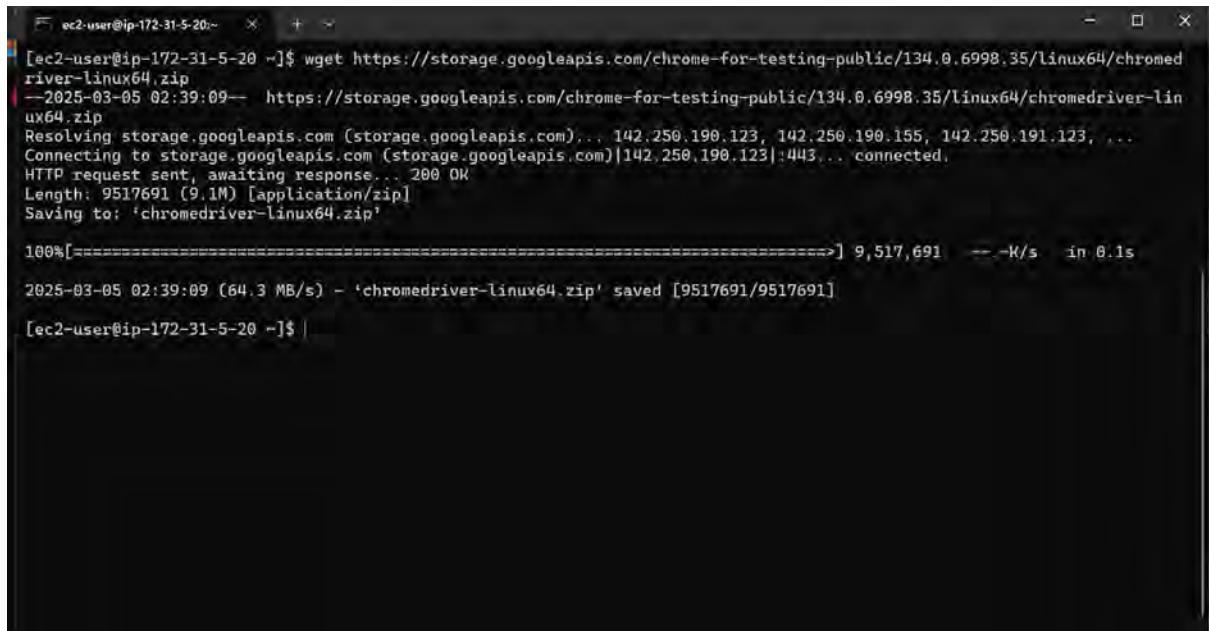
Paso 1: Descargar el instalador desde : <https://googlechromelabs.github.io/chrome-for-testing/#stable>

Paso 2: Ejecutamos el siguiente comando:

```
wget https://storage.googleapis.com/chrome-for-testing-public/134.0.6998.3
→ 5/linux64/chromedriver-linux64.zip
```

Figura 31

Descarga del archivo ChromeDriver desde el repositorio oficial de Google para pruebas automatizadas.



```
ec2-user@ip-172-31-5-20:~$ wget https://storage.googleapis.com/chrome-for-testing-public/134.0.6998.35/linux64/chromedriver-linux64.zip
--2025-03-05 02:39:09-- https://storage.googleapis.com/chrome-for-testing-public/134.0.6998.35/linux64/chromedriver-linux64.zip
Resolving storage.googleapis.com (storage.googleapis.com)... 142.250.190.123, 142.250.190.155, 142.250.191.123, ...
Connecting to storage.googleapis.com (storage.googleapis.com)|142.250.190.123|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9517691 (9.1M) [application/zip]
Saving to: 'chromedriver-linux64.zip'

100%[=====] 9,517,691 -- -K/s in 0.1s

2025-03-05 02:39:09 (64.3 MB/s) - 'chromedriver-linux64.zip' saved [9517691/9517691]

[ec2-user@ip-172-31-5-20:~]$
```

Nota. Elaboración propia.

Paso 3: Mover el archivo a /usr/local/bin/ y luego dar permisos de ejecución.

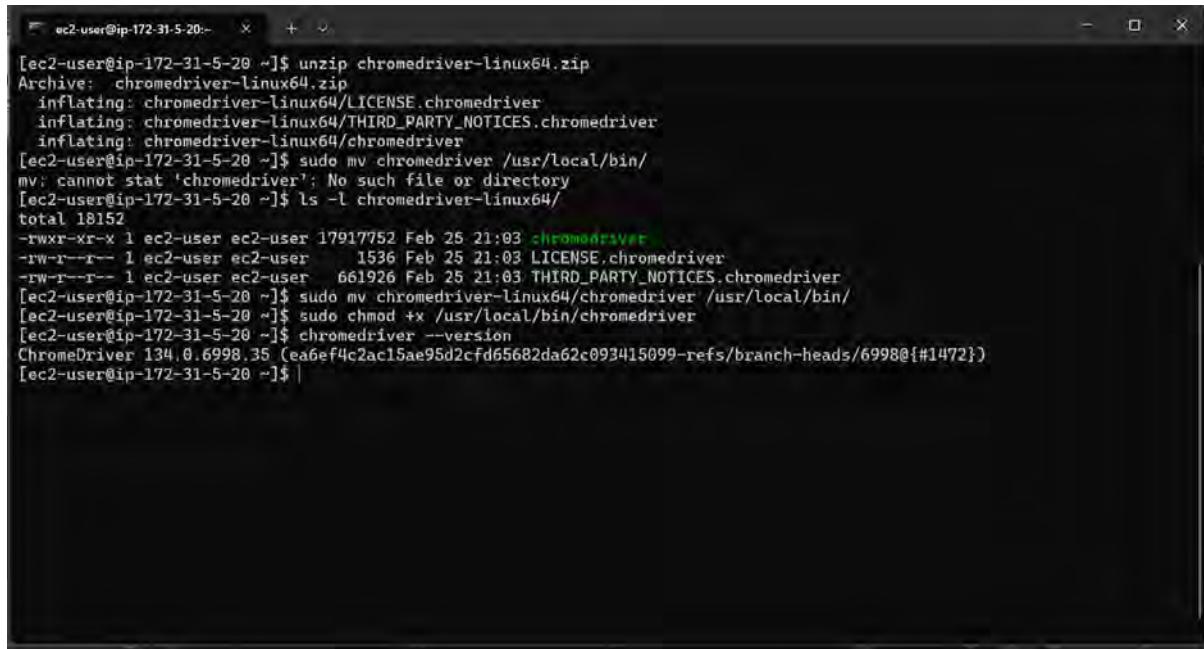
```
sudo mv chromedriver-linux64/chromedriver /usr/local/bin/
sudo chmod +x /usr/local/bin/chromedriver
```

Paso 4: Por último verificamos la instalación

```
chromedriver -version
```

Figura 32

Verificación de la versión instalada de ChromeDriver.



The terminal window shows the following command and its output:

```
ec2-user@ip-172-31-5-20:~]$ unzip chromedriver-linux64.zip
Archive: chromedriver-linux64.zip
  inflating: chromedriver-linux64/LICENSE.chromedriver
  inflating: chromedriver-linux64/THIRD_PARTY_NOTICES.chromedriver
  inflating: chromedriver-linux64/chromedriver
[ec2-user@ip-172-31-5-20 ~]$ sudo mv chromedriver /usr/local/bin/
mv: cannot stat 'chromedriver': No such file or directory
[ec2-user@ip-172-31-5-20 ~]$ ls -l chromedriver-linux64/
total 18152
-rwxr-xr-x 1 ec2-user ec2-user 17917752 Feb 25 21:03 chromedriver
-rw-r--r-- 1 ec2-user ec2-user    1536 Feb 25 21:03 LICENSE.chromedriver
-rw-r--r-- 1 ec2-user ec2-user   661926 Feb 25 21:03 THIRD_PARTY_NOTICES.chromedriver
[ec2-user@ip-172-31-5-20 ~]$ sudo mv chromedriver-linux64/chromedriver /usr/local/bin/
[ec2-user@ip-172-31-5-20 ~]$ sudo chmod +x /usr/local/bin/chromedriver
[ec2-user@ip-172-31-5-20 ~]$ chromedriver --version
ChromeDriver 134.0.6998.35 (ea6ef4c2ac15ae95d2cf655682da62c093415099-refs/branch-heads/6998@{#1472})
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

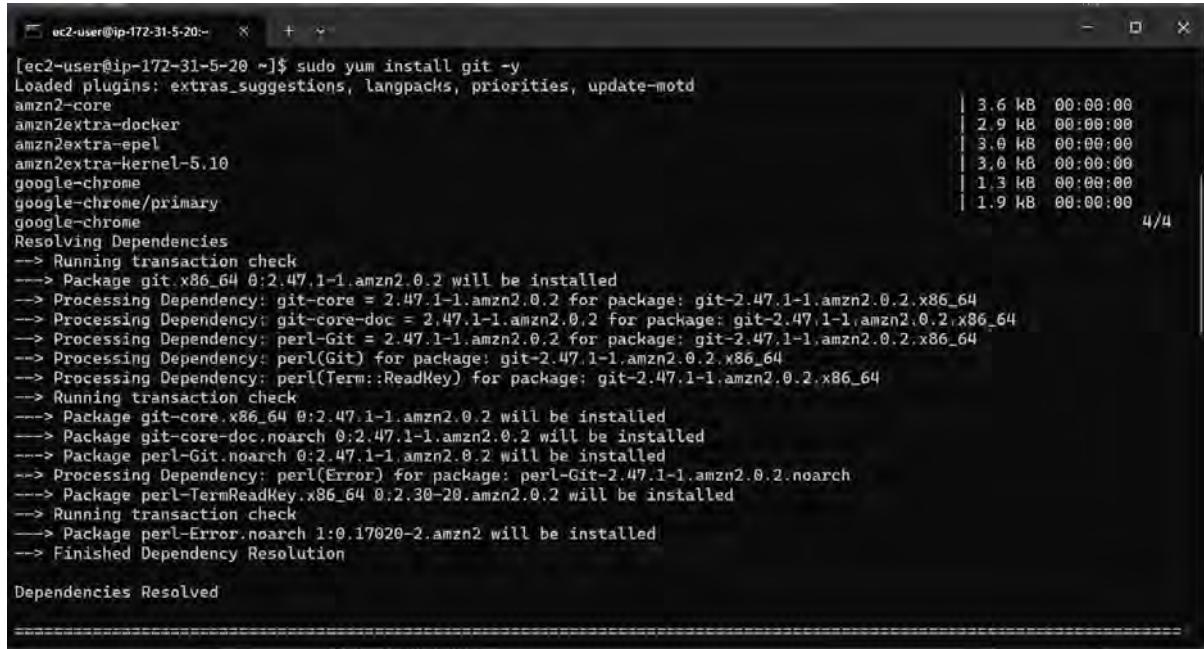
4.9.5. Clonar el repositorio

Paso 1: Instalar git en nuestra instancia.

```
sudo yum install git -y
```

Figura 33

Instalación de Git en Linux usando yum.



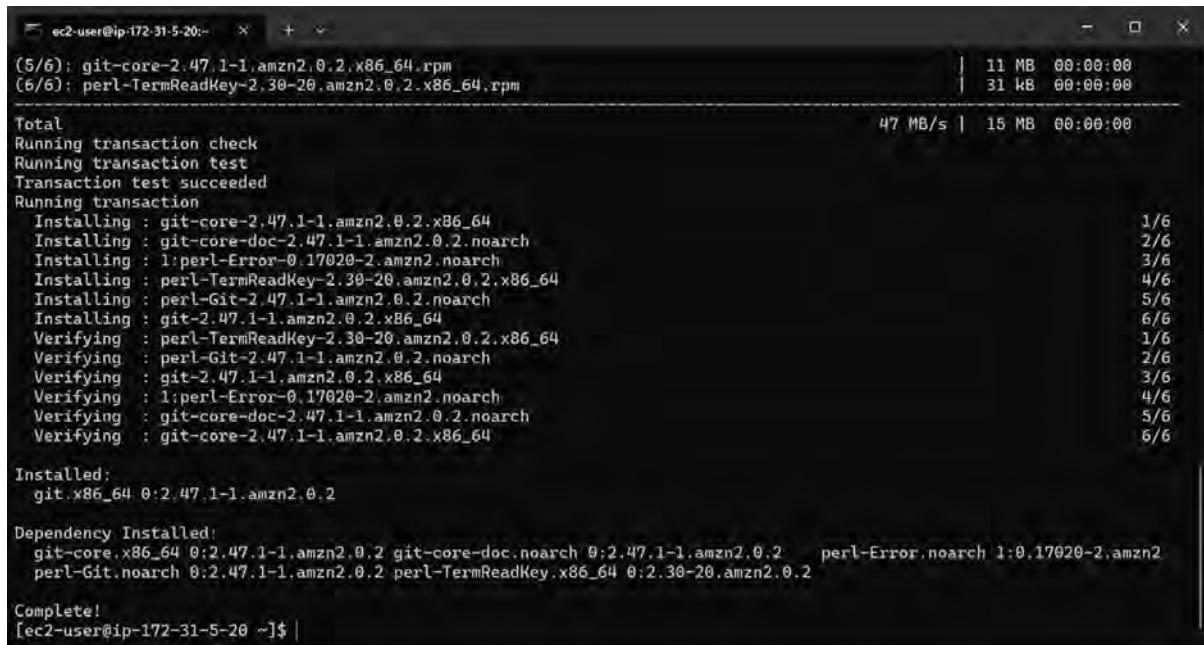
```
[ec2-user@ip-172-31-5-20 ~]$ sudo yum install git -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
amzn2extra-docker
amzn2extra-epel
amzn2extra-kernel-5.10
google-chrome
google-chrome/primary
google-chrome
Resolving Dependencies
--> Running transaction check
--> Package git.x86_64 0:2.47.1-1.amzn2.0.2 will be installed
--> Processing Dependency: git-core = 2.47.1-1.amzn2.0.2 for package: git-2.47.1-1.amzn2.0.2.x86_64
--> Processing Dependency: git-core-doc = 2.47.1-1.amzn2.0.2 for package: git-2.47.1-1.amzn2.0.2.x86_64
--> Processing Dependency: perl-Git = 2.47.1-1.amzn2.0.2 for package: git-2.47.1-1.amzn2.0.2.x86_64
--> Processing Dependency: perl(Git) for package: git-2.47.1-1.amzn2.0.2.x86_64
--> Processing Dependency: perl(Term::ReadKey) for package: git-2.47.1-1.amzn2.0.2.x86_64
--> Running transaction check
--> Package git-core.x86_64 0:2.47.1-1.amzn2.0.2 will be installed
--> Package git-core-doc.noarch 0:2.47.1-1.amzn2.0.2 will be installed
--> Package perl-Git.noarch 0:2.47.1-1.amzn2.0.2 will be installed
--> Processing Dependency: perl(Error) for package: perl-Git-2.47.1-1.amzn2.0.2.noarch
--> Package perl-TermReadKey.x86_64 0:2.30-20.amzn2.0.2 will be installed
--> Running transaction check
--> Package perl-Error.noarch 1:0.17020-2.amzn2 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

Nota. Elaboración propia.

Figura 34

Instalación de Git culminada



```
[5/6]: git-core-2.47.1-1.amzn2.0.2.x86_64.rpm | 11 MB 00:00:00
[6/6]: perl-TermReadKey-2.30-20.amzn2.0.2.x86_64.rpm | 31 kB 00:00:00
Total 47 MB/s | 15 MB 00:00:00

Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : git-core-2.47.1-1.amzn2.0.2.x86_64 1/6
  Installing : git-core-doc-2.47.1-1.amzn2.0.2.noarch 2/6
  Installing : perl-Error-0.17020-2.amzn2.noarch 3/6
  Installing : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64 4/6
  Installing : perl-Git-2.47.1-1.amzn2.0.2.noarch 5/6
  Installing : git-2.47.1-1.amzn2.0.2.x86_64 6/6
  Verifying : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64 1/6
  Verifying : perl-Git-2.47.1-1.amzn2.0.2.noarch 2/6
  Verifying : git-2.47.1-1.amzn2.0.2.x86_64 3/6
  Verifying : perl-Error-0.17020-2.amzn2.noarch 4/6
  Verifying : git-core-doc-2.47.1-1.amzn2.0.2.noarch 5/6
  Verifying : git-core-2.47.1-1.amzn2.0.2.x86_64 6/6

Installed:
  git.x86_64 0:2.47.1-1.amzn2.0.2

Dependency Installed:
  git-core.x86_64 0:2.47.1-1.amzn2.0.2 git-core-doc.noarch 0:2.47.1-1.amzn2.0.2 perl-Error.noarch 1:0.17020-2.amzn2
  perl-Git.noarch 0:2.47.1-1.amzn2.0.2 perl-TermReadKey.x86_64 0:2.30-20.amzn2.0.2

Complete!
[ec2-user@ip-172-31-5-20 ~]$ |
```

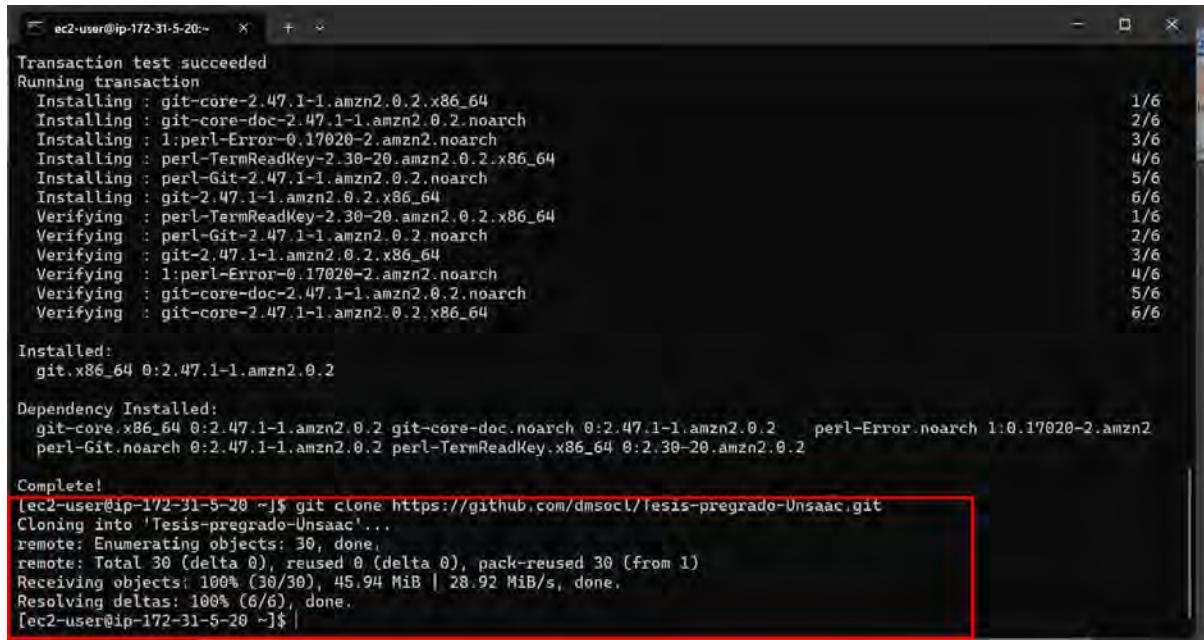
Nota. Elaboración propia.

Paso 2: Clonar el repositorio

```
git clone https://github.com/dmsocl/Tesis-pregrado-Unsaac.git
```

Figura 35

Clonación del repositorio de la tesis desde GitHub mediante git clone.



```

Transaction test succeeded
Running transaction
  Installing : git-core-2.47.1-1.amzn2.0.2.x86_64 1/6
  Installing : git-core-doc-2.47.1-1.amzn2.0.2.noarch 2/6
  Installing : perl-Error-0.17020-2.amzn2.0.noarch 3/6
  Installing : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64 4/6
  Installing : perl-Git-2.47.1-1.amzn2.0.2.noarch 5/6
  Installing : git-2.47.1-1.amzn2.0.2.x86_64 6/6
  Verifying  : perl-TermReadKey-2.30-20.amzn2.0.2.x86_64 1/6
  Verifying  : perl-Git-2.47.1-1.amzn2.0.2.noarch 2/6
  Verifying  : git-2.47.1-1.amzn2.0.2.x86_64 3/6
  Verifying  : perl-Error-0.17020-2.amzn2.0.noarch 4/6
  Verifying  : git-core-doc-2.47.1-1.amzn2.0.2.noarch 5/6
  Verifying  : git-core-2.47.1-1.amzn2.0.2.x86_64 6/6

Installed:
  git.x86_64 0:2.47.1-1.amzn2.0.2

Dependency Installed:
  git-core.x86_64 0:2.47.1-1.amzn2.0.2 git-core-doc.noarch 0:2.47.1-1.amzn2.0.2 perl-Error.noarch 1:0.17020-2.amzn2
  perl-Git.noarch 0:2.47.1-1.amzn2.0.2 perl-TermReadKey.x86_64 0:2.30-20.amzn2.0.2

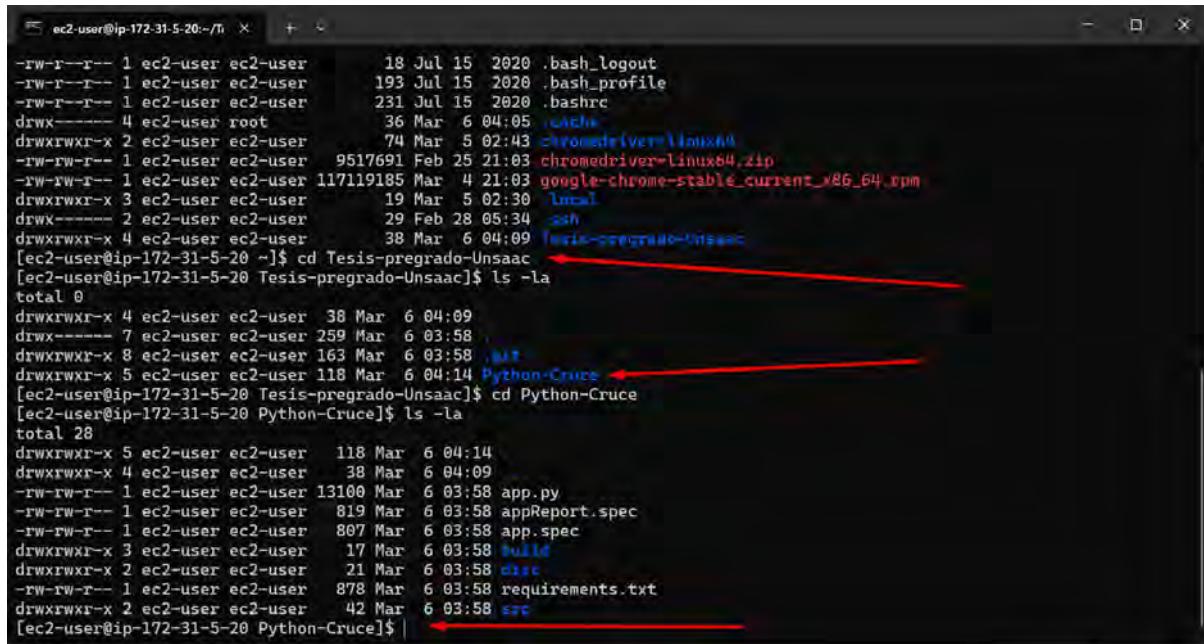
Complete!
[ec2-user@ip-172-31-5-20 ~]$ git clone https://github.com/dmsocl/Tesis-pregrado-Unsaac.git
Cloning into 'Tesis-pregrado-Unsaac'...
remote: Enumerating objects: 30, done.
remote: Total 30 (delta 0), reused 0 (delta 0), pack-reused 30 (from 1)
Receiving objects: 100% (30/30), 45.94 MiB | 28.92 MiB/s, done.
Resolving deltas: 100% (6/6), done.
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

Paso 3: Nos dirigimos a la carpeta de nuestro proyecto e instalamos el entorno virtual y luego activamos el entorno virtual, cuando veamos el prefijo “venv” significa que ya está activado el entorno virtual.

Figura 36

Ingreso al directorio del proyecto clonado.

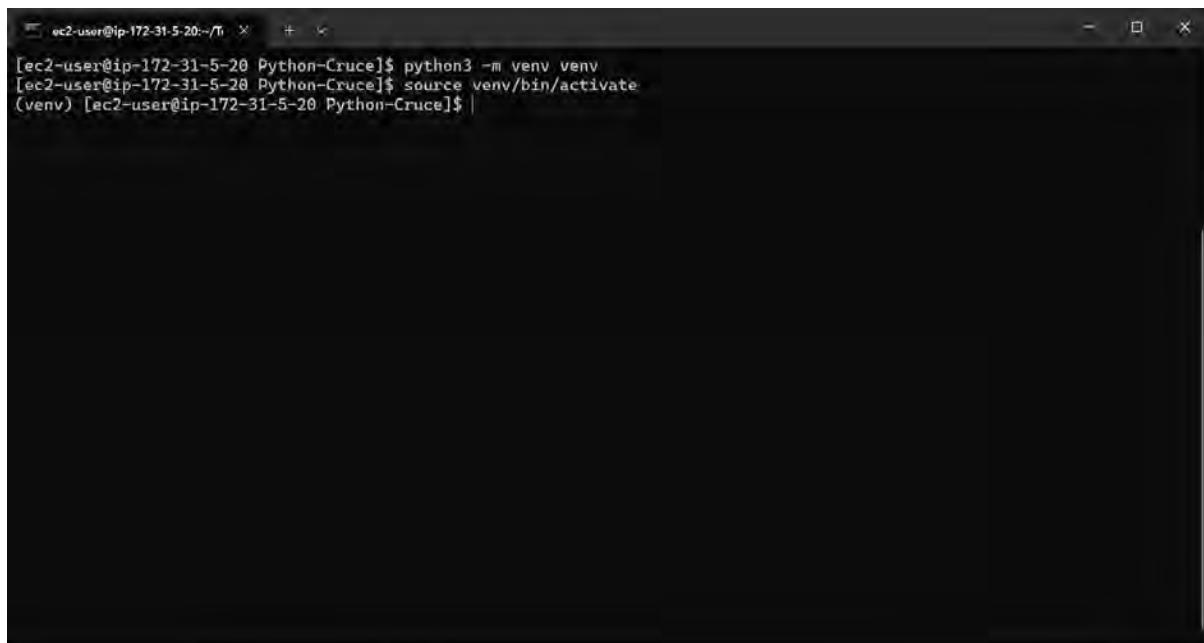


```
ec2-user@ip-172-31-5-20:~/Tesis-pregrado-Unsaac$ ls -la
-rw-r--r-- 1 ec2-user ec2-user 18 Jul 15 2020 .bash_logout
-rw-r--r-- 1 ec2-user ec2-user 193 Jul 15 2020 .bash_profile
-rw-r--r-- 1 ec2-user ec2-user 231 Jul 15 2020 .bashrc
drwxr--r-- 4 ec2-user root 36 Mar 6 04:05 .cache
drwxrwxr-x 2 ec2-user ec2-user 74 Mar 5 02:43 chromedriver-linux64
-rw-rw-r-- 1 ec2-user ec2-user 9517691 Feb 25 21:03 chromedriver-linux64.zip
-rw-rw-r-- 1 ec2-user ec2-user 117119185 Mar 4 21:03 google-chrome-stable_current_x86_64.rpm
drwxrwxr-x 3 ec2-user ec2-user 19 Mar 5 02:30 .local
drwxr--r-- 2 ec2-user ec2-user 29 Feb 28 05:34 .ssh
drwxrwxr-x 4 ec2-user ec2-user 38 Mar 6 04:09 lexis-cregrado-Unsaac
[ec2-user@ip-172-31-5-20 ~]$ cd Tesis-pregrado-Unsaac
[ec2-user@ip-172-31-5-20 Tesis-pregrado-Unsaac]$ ls -la
total 0
drwxrwxr-x 4 ec2-user ec2-user 38 Mar 6 04:09
drwxr--r-- 7 ec2-user ec2-user 259 Mar 6 03:58 .
drwxrwxr-x 8 ec2-user ec2-user 163 Mar 6 03:58 .git
drwxrwxr-x 5 ec2-user ec2-user 118 Mar 6 04:14 Python-Cruce
[ec2-user@ip-172-31-5-20 Tesis-pregrado-Unsaac]$ cd Python-Cruce
[ec2-user@ip-172-31-5-20 Python-Cruce]$ ls -la
total 28
drwxrwxr-x 5 ec2-user ec2-user 118 Mar 6 04:14
drwxrwxr-x 4 ec2-user ec2-user 38 Mar 6 04:09
-rw-rw-r-- 1 ec2-user ec2-user 13100 Mar 6 03:58 app.py
-rw-rw-r-- 1 ec2-user ec2-user 819 Mar 6 03:58 appReport.spec
-rw-rw-r-- 1 ec2-user ec2-user 807 Mar 6 03:58 app.spec
drwxrwxr-x 3 ec2-user ec2-user 17 Mar 6 03:58 build
drwxrwxr-x 2 ec2-user ec2-user 21 Mar 6 03:58 dist
-rw-rw-r-- 1 ec2-user ec2-user 878 Mar 6 03:58 requirements.txt
drwxrwxr-x 2 ec2-user ec2-user 42 Mar 6 03:58 src
[ec2-user@ip-172-31-5-20 Python-Cruce]$ |
```

Nota. Elaboración propia.

Figura 37

Creación y activación del entorno virtual en el proyecto.



```
ec2-user@ip-172-31-5-20:~/Tesis-pregrado-Unsaac$ python3 -m venv venv
[ec2-user@ip-172-31-5-20 Python-Cruce]$ source venv/bin/activate
(venv) [ec2-user@ip-172-31-5-20 Python-Cruce]$ |
```

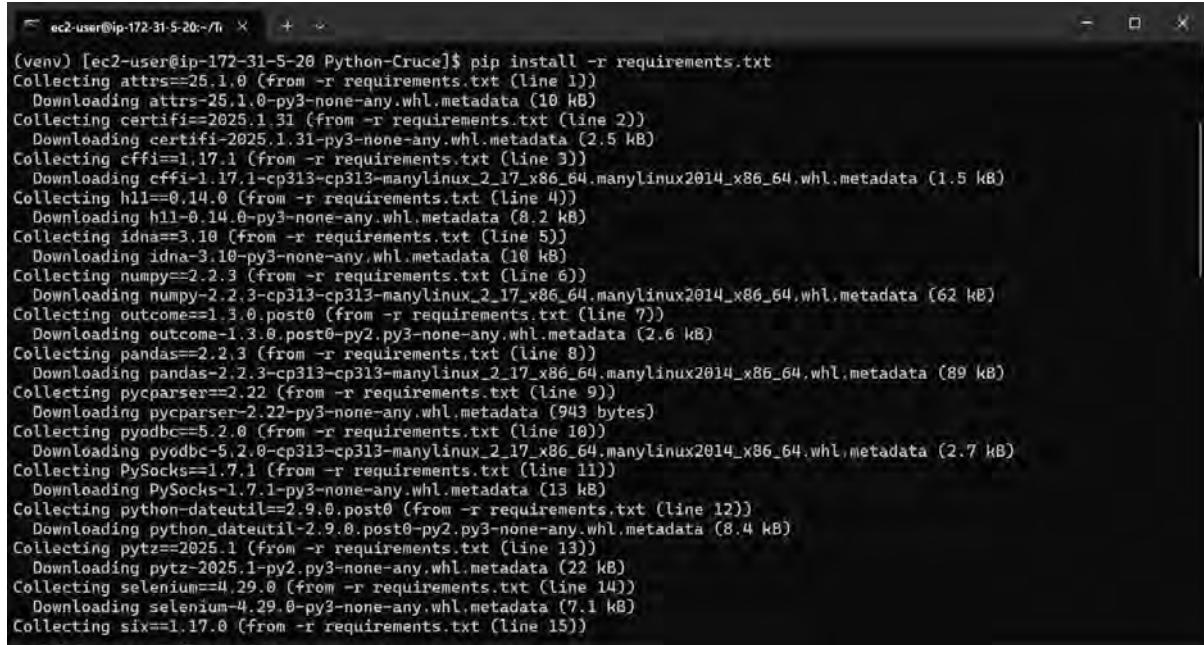
Nota. Elaboración propia.

Paso 4: Instalar todas las dependencias que necesita el proyecto.

```
pip install -r requirements.txt
```

Figura 38

Instalación de dependencias del proyecto con pip.

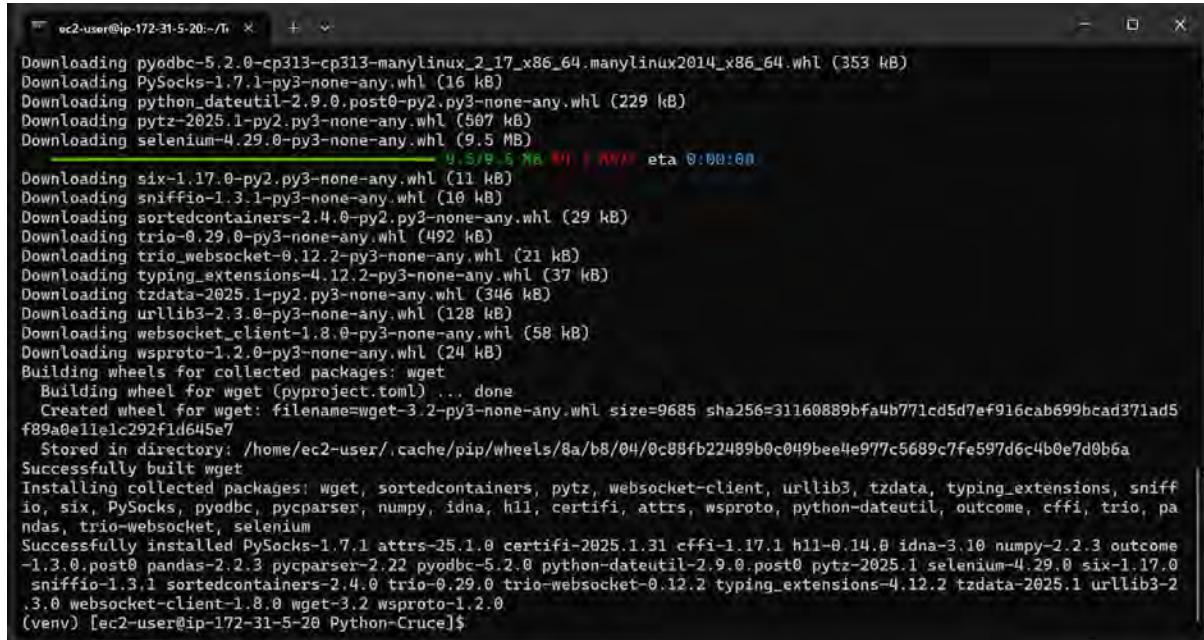


```
(venv) [ec2-user@ip-172-31-5-20 Python-Cruce]$ pip install -r requirements.txt
Collecting attrs==25.1.0 (from -r requirements.txt (line 1))
  Downloading attrs-25.1.0-py3-none-any.whl.metadata (10 kB)
Collecting certifi==2025.1.31 (from -r requirements.txt (line 2))
  Downloading certifi-2025.1.31-py3-none-any.whl.metadata (2.5 kB)
Collecting cffi==1.17.1 (from -r requirements.txt (line 3))
  Downloading cffi-1.17.1-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting h11==0.14.0 (from -r requirements.txt (line 4))
  Downloading h11-0.14.0-py3-none-any.whl.metadata (8.2 kB)
Collecting idna==3.10 (from -r requirements.txt (line 5))
  Downloading idna-3.10-py3-none-any.whl.metadata (10 kB)
Collecting numpy==2.2.3 (from -r requirements.txt (line 6))
  Downloading numpy-2.2.3-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (62 kB)
Collecting outcome==1.3.0.post0 (from -r requirements.txt (line 7))
  Downloading outcome-1.3.0.post0-py2.py3-none-any.whl.metadata (2.6 kB)
Collecting pandas==2.2.3 (from -r requirements.txt (line 8))
  Downloading pandas-2.2.3-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (89 kB)
Collecting pycparser==2.22 (from -r requirements.txt (line 9))
  Downloading pycparser-2.22-py3-none-any.whl.metadata (943 bytes)
Collecting pyodbc==5.2.0 (from -r requirements.txt (line 10))
  Downloading pyodbc-5.2.0-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (2.7 kB)
Collecting PySocks==1.7.1 (from -r requirements.txt (line 11))
  Downloading PySocks-1.7.1-py3-none-any.whl.metadata (13 kB)
Collecting python-dateutil==2.9.0.post0 (from -r requirements.txt (line 12))
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting pytz==2025.1 (from -r requirements.txt (line 13))
  Downloading pytz-2025.1-py2.py3-none-any.whl.metadata (22 kB)
Collecting selenium==4.29.0 (from -r requirements.txt (line 14))
  Downloading selenium-4.29.0-py3-none-any.whl.metadata (7.1 kB)
Collecting six==1.17.0 (from -r requirements.txt (line 15))
```

Nota. Elaboración propia.

Figura 39

Instalación de dependencias del proyecto con pip culminada.



```
ec2-user@ip-172-31-5-20:~/. Tr * + ~
Downloading pyodbc-5.2.0-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (353 kB)
Downloading PySocks-1.7.1-py3-none-any.whl (16 kB)
Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
Downloading pytz-2025.1-py2.py3-none-any.whl (507 kB)
Downloading selenium-4.29.0-py3-none-any.whl (9.5 MB)
  Downloading selenium-4.29.0-py3-none-any.whl (9.5 MB) 0.5/9.5 MB 80% [=====] eta 0:00:08
Downloading six-1.17.0-py2.py3-none-any.whl (11 kB)
Downloading sniffio-1.3.1-py3-none-any.whl (10 kB)
Downloading sortedcontainers-2.4.0-py2.py3-none-any.whl (29 kB)
Downloading trio-0.29.0-py3-none-any.whl (492 kB)
Downloading trio_websocket-0.12.2-py3-none-any.whl (21 kB)
Downloading typing_extensions-4.12.2-py3-none-any.whl (37 kB)
Downloading tzdata-2025.1-py2.py3-none-any.whl (346 kB)
Downloading urllib3-2.3.0-py3-none-any.whl (128 kB)
Downloading websocket_client-1.8.0-py3-none-any.whl (58 kB)
Downloading wsproto-1.2.0-py3-none-any.whl (24 kB)
Building wheels for collected packages: wget
  Building wheel for wget (pyproject.toml) ... done
  Created wheel for wget: filename=wget-3.2-py3-none-any.whl size=9685 sha256=31160889bfa4b771cd5d7ef916cab699bcad371ad5
f89a0e11e1c292f1d645e7
  Stored in directory: /home/ec2-user/.cache/pip/wheels/8a/b8/04/0c88fb22489b0c049bee4e977c5689c7fe597d6c4b0e7d0b6a
Successfully built wget
Installing collected packages: wget, sortedcontainers, pytz, websocket-client, urllib3, tzdata, typing_extensions, sniffio, six, PySocks, pyodbc, pycparser, numpy, idna, h11, certifi, attrs, wsproto, python-dateutil, outcome, cffi, trio, pandas, trio-websocket, selenium
Successfully installed PySocks-1.7.1 attrs-25.1.0 certifi-2025.1.31 cffi-1.17.1 h11-0.14.0 idna-3.10 numpy-2.2.3 outcome-1.3.0.post0 pandas-2.2.3 pycparser-2.22 pyodbc-5.2.0 python-dateutil-2.9.0.post0 pytz-2025.1 selenium-4.29.0 six-1.17.0 sniffio-1.3.1 sortedcontainers-2.4.0 trio-0.29.0 trio_websocket-0.12.2 typing_extensions-4.12.2 tzdata-2025.1 urllib3-2.3.0 websocket-client-1.8.0 wget-3.2 wsproto-1.2.0
(venv) [ec2-user@ip-172-31-5-20 Python-Cruce]$
```

Nota. Elaboración propia.

4.9.6. Automatizar con Cron

Paso 1: Instalar cron.

```
sudo yum install cronie -y
```

Figura 40

Instalación del servicio de tareas programadas cronie en Linux.

```
ec2-user@ip-172-31-5-20:~ % Microsoft Windows [Versión 10.0.26100.3194]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user@18.222.14.69
ssh: connect to host 18.222.14.69 port 22: Connection timed out

C:\Users\dmsoc\Downloads>ssh -i clave.pem ec2-user@18.222.14.69
Last login: Thu Mar  6 05:41:48 2025 from 190.238.57.214
      #
      ' \_ #####_      Amazon Linux 2
      ~ \_#####\_
      ~ \###|      AL2 End of Life is 2026-06-30.
      ~ \#/ __
      ~ \~' ' ->
      ~ \_ /      A newer version of Amazon Linux is available!
      ~ \_ /      Amazon Linux 2023, GA and supported until 2028-03-15.
      ~ \_ /      https://aws.amazon.com/linux/amazon-linux-2023/
      ~ \_ /      https://aws.amazon.com/linux/amazon-linux-2023/
[ec2-user@ip-172-31-5-20 ~]$ sudo yum install cronie -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
(1/2): amzn2-core/2/x86_64/updateinfo
(2/2): amzn2-core/2/x86_64/primary_db
Package cronie-1.4.11-23.amzn2.x86_64 already installed and latest version
Nothing to do
[ec2-user@ip-172-31-5-20 ~]$ |
```

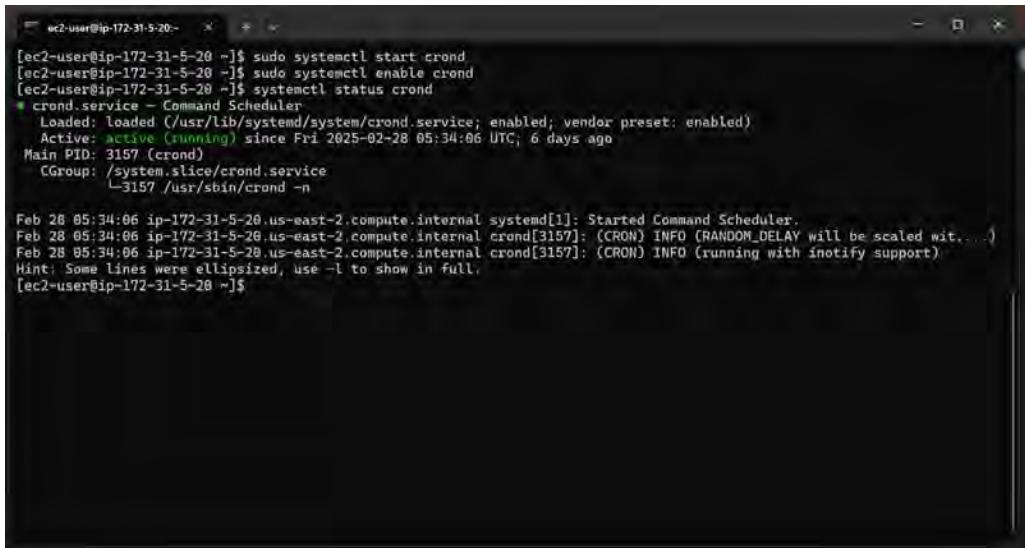
Nota. Elaboración propia.

Paso 2: Iniciar y habilitar el servicio y verificamos que este corriendo

```
sudo systemctl start crond  
sudo systemctl enable crond  
systemctl status crond
```

Figura 41

Activación y verificación del servicio crond para tareas programadas.



```
[ec2-user@ip-172-31-5-20- ~]$ sudo systemctl start crond
[ec2-user@ip-172-31-5-20- ~]$ sudo systemctl enable crond
[ec2-user@ip-172-31-5-20- ~]$ systemctl status crond
● crond.service - Command Scheduler
   Loaded: loaded (/usr/lib/systemd/system/crond.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2025-02-28 05:34:06 UTC; 6 days ago
       Main PID: 3157 (crond)
          CGroup: /system.slice/crond.service
                  └─3157 /usr/sbin/crond -n

Feb 28 05:34:06 ip-172-31-5-20.us-east-2.compute.internal systemd[1]: Started Command Scheduler.
Feb 28 05:34:06 ip-172-31-5-20.us-east-2.compute.internal crond[3157]: (CRON) INFO (RANDOM_DELAY will be scaled with...)
Feb 28 05:34:06 ip-172-31-5-20.us-east-2.compute.internal crond[3157]: (CRON) INFO (running with inotify support)
Hint: Some lines were ellipsized, use -l to show in full.
[ec2-user@ip-172-31-5-20- ~]$
```

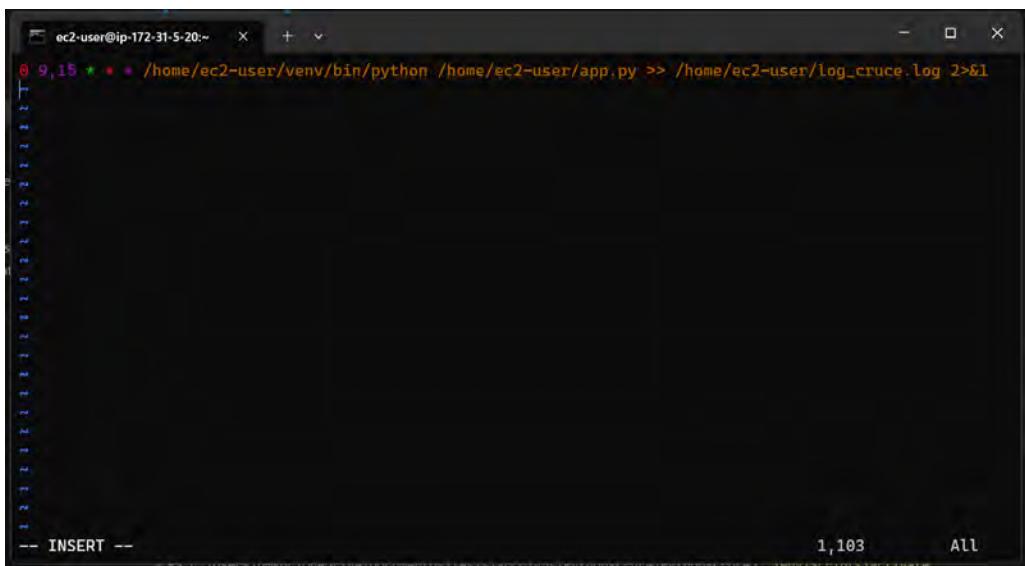
Nota. Elaboración propia.

Paso 3: Edita crontab y añadir una entrada para ejecutar diariamente 6am y 3pm.

```
crontab -e
0 9,15 * * * /home/ec2-user/venv/bin/python /home/ec2-user/tu_script.py >>
↪ /home/ec2-user/log_cruce.log 2>&1
```

Figura 42

Edición del archivo crontab para programar tareas automáticas.



```
[ec2-user@ip-172-31-5-20- ~]$ 0 9,15 * * * /home/ec2-user/venv/bin/python /home/ec2-user/app.py >> /home/ec2-user/log_cruce.log 2>&1
```

Nota. Elaboración propia.

Paso 4: Esto hará que el script se ejecute a las 9:00 AM y a las 3:00 PM todos los días.

- 0 → Minuto exacto 0.
- 9,15 → Horas (9 AM y 3 PM en formato 24 horas).
- *** → Todos los días del mes, todos los meses, todos los días de la semana.
- >> log_cruce.log 2>&1 → Guarda el registro de ejecución y errores en un archivo log_cruce.log.

Figura 43

Guardar el archivo crontab



```
ec2-user@ip-172-31-5-20:~$ crontab -e
no crontab for ec2-user - using an empty one
crontab: installing new crontab
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

Paso 5: Con crontab -l verificamos que se haya añadido correctamente nuestra entrada.

Figura 44

Verificación de tareas programadas con crontab -l.



```
ec2-user@ip-172-31-5-20:~$ crontab -l
0 9,15 * * * /home/ec2-user/venv/bin/python /home/ec2-user/app.py >> /home/ec2-user/log_cruce.log 2>&
[ec2-user@ip-172-31-5-20 ~]$
```

Nota. Elaboración propia.

Capítulo 5

Análisis y discusión de resultados

5.1. Resultados

- Se implementó un sistema de gestión e integración de información contable y financiera de los tres sistemas involucrados: RMS, Acepta y SAP. Basado en RPA, ETL Y AWS.
- Se erradicó el factor humano de error. Lo cual lleva a una mayor eficiencia en la gestión de los datos y procesos financieros.
- El sistema facilita la integración de datos de diversas fuentes, lo que permitirá a Topitop obtener una vista más completa del estado de sus documentos contables.
- AWS proporcionará una plataforma escalable y segura para el almacenamiento y el acceso a datos en tiempo real, lo que permitirá tomar decisiones basadas en información actualizada.
- Mayor cumplimiento normativo respecto a SUNAT, reduciendo así los riesgos legales y financieros.
- El tiempo de se redujo considerablemente, ahora todo el proceso demora de 40 minutos a 1 hora.

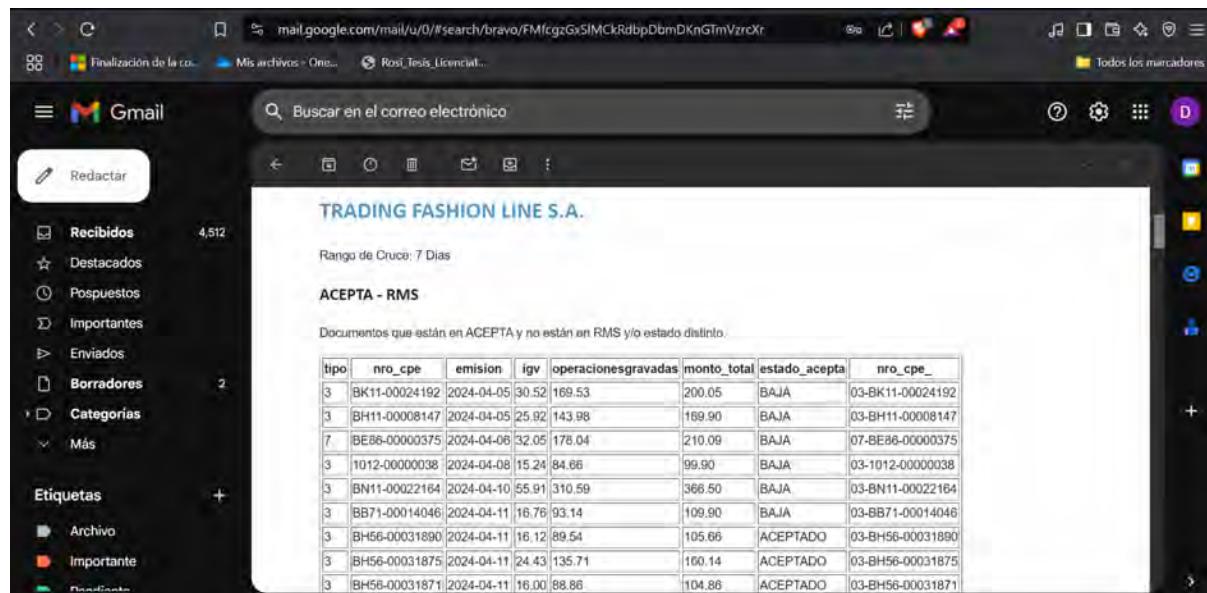
En conclusión, la implementación exitosa del Sistema de Gestión e Integración de Información Contable y Financiera basado en RPA, ETL y AWS para Topitop condujo a una mejora significativa en la eficiencia, precisión y capacidad de toma de decisiones de la empresa, lo que a su vez impulsa su competitividad y éxito a largo plazo en el mercado.

5.2. Verificación del comportamiento del sistema

A continuación, se muestra la alerta enviada por correo electrónico, la cual detalla los documentos contables que presentan errores. De esta manera, el colaborador del área de TI puede corregir dichos documentos, asegurando que la información esté sincronizada y actualizada en todos los sistemas correspondientes.

Figura 45

Captura de Documentos Procesados en ACCEPTA y no en RMS.



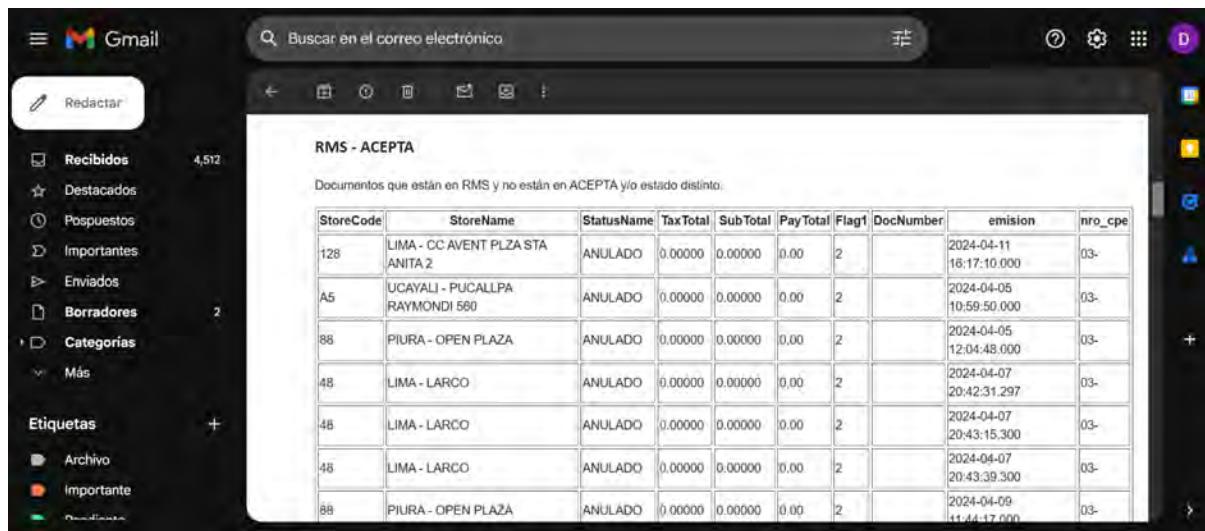
The screenshot shows a Gmail inbox with the search bar set to 'Buscar en el correo electrónico'. A message from 'TRADING FASHION LINE S.A.' is displayed, titled 'ACEPTA - RMS'. The message content is: 'Documentos que están en ACCEPTA y no están en RMS y/o estado distinto.' Below this, a table lists 13 documents with their details:

tipo	nro_cpe	emision	igv	operacionesgravadas	monto_total	estado_acepta	nro_cpe_
3	BK11-00024192	2024-04-05	30.52	169.53	200.05	BAJA	03-BK11-00024192
3	BH11-00008147	2024-04-05	25.92	143.98	169.90	BAJA	03-BH11-00008147
7	BE86-00000375	2024-04-06	32.05	178.04	210.09	BAJA	07-BE86-00000375
3	1012-00000038	2024-04-08	15.24	84.66	99.90	BAJA	03-1012-00000038
3	BN11-00022164	2024-04-10	55.91	310.59	366.50	BAJA	03-BN11-00022164
3	BB71-00014046	2024-04-11	16.76	93.14	109.90	BAJA	03-BB71-00014046
3	BH56-00031890	2024-04-11	16.12	89.54	105.66	ACEPTADO	03-BH56-00031890
3	BH56-00031875	2024-04-11	24.43	135.71	160.14	ACEPTADO	03-BH56-00031875
3	BH56-00031871	2024-04-11	16.00	88.86	104.86	ACEPTADO	03-BH56-00031871

Nota. Elaboración propia.

Figura 46

Captura de Documentos Procesados en RMS y no en ACEPTA.



RMS - ACEPTA

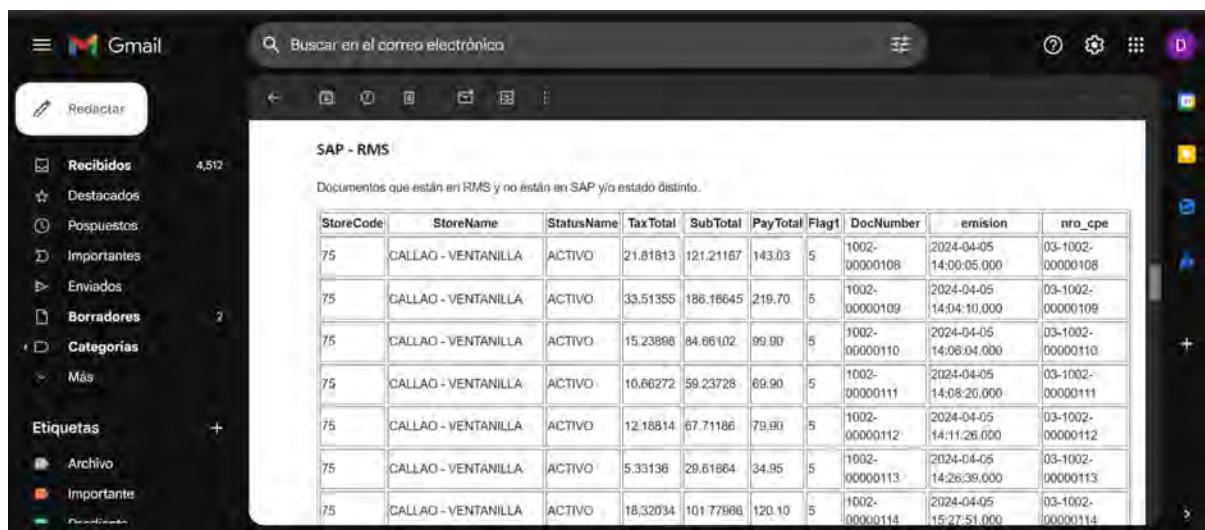
Documentos que están en RMS y no están en ACEPTA y/o estado distinto:

StoreCode	StoreName	StatusName	TaxTotal	SubTotal	PayTotal	Flag1	DocNumber	emision	nro_cpe
128	LIMA - CC AVVENT PLZA STA ANITA 2	ANULADO	0.00000	0.00000	0.00	2		2024-04-11 16:17:10.000	03-
A5	UCAYALI - PUCALLPA RAYMONDI 580	ANULADO	0.00000	0.00000	0.00	2		2024-04-05 10:59:50.000	03-
88	PIURA - OPEN PLAZA	ANULADO	0.00000	0.00000	0.00	2		2024-04-05 12:04:48.000	03-
48	LIMA - LARCO	ANULADO	0.00000	0.00000	0.00	2		2024-04-07 20:42:31.297	03-
48	LIMA - LARCO	ANULADO	0.00000	0.00000	0.00	2		2024-04-07 20:43:15.300	03-
48	LIMA - LARCO	ANULADO	0.00000	0.00000	0.00	2		2024-04-07 20:43:39.300	03-
88	PIURA - OPEN PLAZA	ANULADO	0.00000	0.00000	0.00	2		2024-04-09 11:44:17.000	03-

Nota. Elaboración propia.

Figura 47

Captura de Documentos Procesados en SAP y no en RMS.



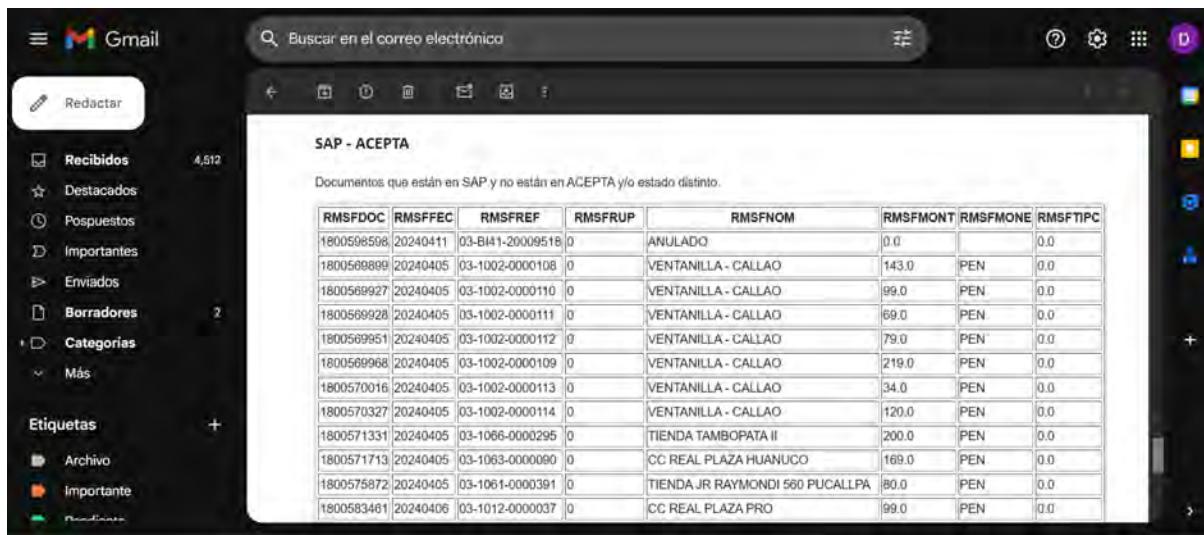
SAP - RMS

Documentos que están en SAP y no están en RMS y/o estado distinto:

StoreCode	StoreName	StatusName	TaxTotal	SubTotal	PayTotal	Flag1	DocNumber	emision	nro_cpe
75	CALLAO - VENTANILLA	ACTIVO	21.81813	121.21167	143.03	5	1002-00000108	2024-04-05 14:00:05.000	03-1002-00000108
75	CALLAO - VENTANILLA	ACTIVO	33.51355	186.16645	219.70	5	1002-00000109	2024-04-05 14:04:10.000	03-1002-00000109
75	CALLAO - VENTANILLA	ACTIVO	15.23898	84.86102	99.90	5	1002-00000110	2024-04-05 14:06:04.000	03-1002-00000110
75	CALLAO - VENTANILLA	ACTIVO	10.66272	59.23728	69.90	5	1002-00000111	2024-04-05 14:08:20.000	03-1002-00000111
75	CALLAO - VENTANILLA	ACTIVO	12.18814	67.71186	79.00	5	1002-00000112	2024-04-05 14:11:26.000	03-1002-00000112
75	CALLAO - VENTANILLA	ACTIVO	5.33138	29.61664	34.95	5	1002-00000113	2024-04-05 14:26:39.000	03-1002-00000113
75	CALLAO - VENTANILLA	ACTIVO	18.32034	101.77986	120.10	5	1002-00000114	2024-04-05 15:27:51.000	03-1002-00000114

Nota. Elaboración propia.

Figura 48
Captura de Documentos Procesados en SAP y no en ACEPTA.



RMSFDOC	RMSFFEC	RMSFREF	RMSFRUP	RMSFNOM	RMSFMONT	RMSFMONE	RMSFTIPC
1800598598	20240411	03-BI41-20009518	0	ANULADO	0.0		0.0
1800569899	20240405	03-1002-0000108	0	VENTANILLA - CALLAO	143.0	PEN	0.0
1800569927	20240405	03-1002-0000110	0	VENTANILLA - CALLAO	99.0	PEN	0.0
1800569928	20240405	03-1002-0000111	0	VENTANILLA - CALLAO	69.0	PEN	0.0
1800569951	20240405	03-1002-0000112	0	VENTANILLA - CALLAO	79.0	PEN	0.0
1800569968	20240405	03-1002-0000109	0	VENTANILLA - CALLAO	219.0	PEN	0.0
1800570016	20240405	03-1002-0000113	0	VENTANILLA - CALLAO	34.0	PEN	0.0
1800570327	20240405	03-1002-0000114	0	VENTANILLA - CALLAO	120.0	PEN	0.0
1800571331	20240405	03-1066-0000295	0	TIENDA TAMBOPATA II	200.0	PEN	0.0
1800571713	20240405	03-1063-0000090	0	CC REAL PLAZA HUANUCO	169.0	PEN	0.0
1800575872	20240405	03-1061-0000391	0	TIENDA JR RAYMONDI 560 PUCALLPA	80.0	PEN	0.0
1800583461	20240406	03-1012-0000037	0	CC REAL PLAZA PRO	99.0	PEN	0.0

Nota. Elaboración propia.

5.3. Análisis de resultados respecto a los objetivos

Durante el desarrollo del proyecto, se abordaron de manera efectiva los objetivos planteados. A continuación, se presenta el análisis de resultados por objetivo, considerando el trabajo realizado, el resultado obtenido y su evidencia.

Objetivo 1:

- **Acción:** Se realizó la integración y validación de la información contable y financiera para la conciliación.
- **Resultado:** Se eliminaron las inconsistencias detectadas, incrementando la confiabilidad de la información procesada.
- **Evidencia:** Reportes/bitácoras del proceso de conciliación y validaciones aplicadas.

Objetivo 2:

- **Acción:** Se automatizaron tareas repetitivas del flujo de conciliación para reducir tiempos operativos.
- **Resultado:** El tiempo de conciliación se redujo de aproximadamente 3 horas a un rango de 40 minutos a 1 hora.
- **Evidencia:** Comparativo de tiempos antes y después del sistema.

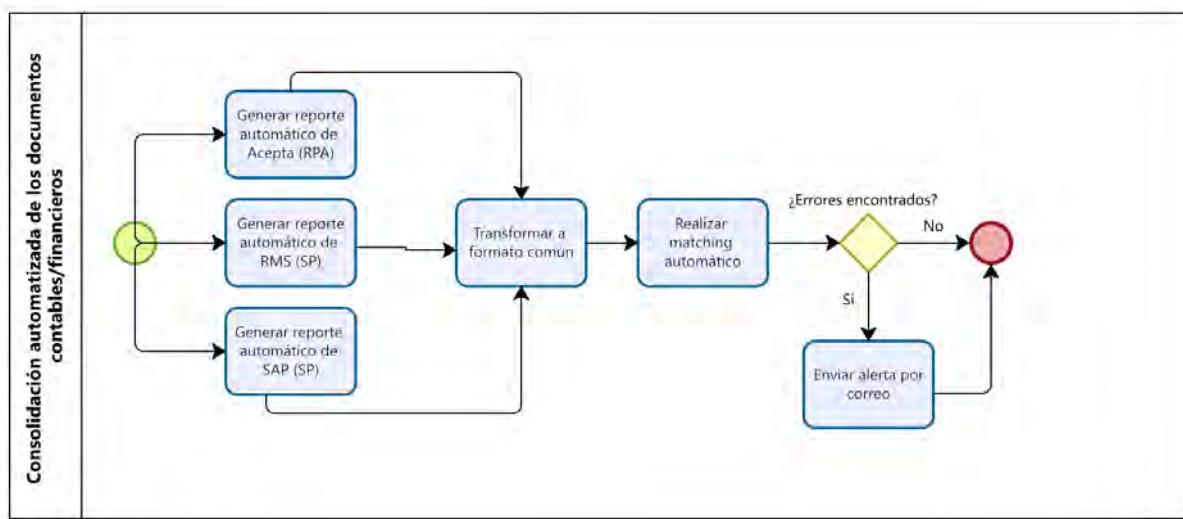
Objetivo 3:

- **Acción:** Se incorporó un mecanismo de alertas ante errores durante el proceso.
- **Resultado:** Se generaron alertas automáticas y se registró la trazabilidad de los eventos.
- **Evidencia:** Correos de alerta y registros de ejecución (logs) del sistema.

Objetivo 4:

- **Acción:** Se validó el sistema mediante interacción con usuarios finales.
- **Resultado:** Los usuarios reportaron usabilidad y estabilidad del sistema y de los documentos financieros reportados.
- **Evidencia:** Resultados de validación/encuesta o actas de pruebas con usuarios.

Figura 49
Procesos automatizados



Nota. *Elaboración propia.*

Tabla 5
Detalle del tiempo invertido en los procedimientos automatizados

Procedimiento	Tiempo (minutos)	Período (días)
Generar reporte de RMS	5	7
Generar reporte de ACEPTA	25	7
Generar reporte de SAP	5	7
Realizar comparación de los documentos	5	7
Total	40	—

Nota. *Elaboración propia.*

Tabla 6

Resumen comparativo antes vs. después con ahorro absoluto y relativo

Procedimiento	Antes (min)	Después (min)	Δ (min)	Reducción (%)
Generar reporte de RMS	40	5	35	87.5
Generar reporte de ACCEPTA	40	25	15	37.5
Generar reporte de SAP	40	5	35	87.5
Realizar comparación de documentos	60	5	55	91.7
Total	180	40	140	77.8

Nota. Periodicidad: antes, reportes diarios (rango de 1 día); después, reportes semanales (rango de 7 días).

Conclusiones

Los resultados obtenidos a partir de la implementación del sistema confirman el cumplimiento de los objetivos establecidos, al evidenciar mejoras sustanciales en la eficiencia operativa, la eliminación total de los errores manuales en los procesos contables y financieros, así como una disminución significativa de los riesgos asociados al incumplimiento tributario. De acuerdo con el proceso desarrollado, se extraen las siguientes conclusiones:

1. La integración automatizada entre RMS, SAP y ACCEPTA eliminó los errores manuales de conciliación contable y financiera, asegurando mayor confiabilidad y consistencia en la información procesada.
2. La automatización del flujo redujo los retrasos operativos, disminuyendo el tiempo de conciliación.
3. La estandarización de datos y la incorporación de alertas automáticas minimizaron el riesgo de incumplimiento tributario, garantizando el envío oportuno y correcto de la información a la SUNAT.
4. La validación del sistema, mediante la interacción con los usuarios finales, confirmó la usabilidad, estabilidad y el impacto positivo del sistema en la gestión diaria.

Recomendaciones

A partir del desarrollo e implementación del sistema propuesto, se recomiendan las siguientes acciones para asegurar su sostenibilidad, mejora continua y posibilidad de expansión:

- **Capacitación del personal clave:** Realizar capacitaciones periódicas al equipo técnico y contable sobre el uso del sistema y sobre cambios relevantes en RPA, ETL y AWS, a fin de asegurar continuidad operativa.
- **Monitoreo y mantenimiento del sistema:** Implementar monitoreo continuo de ejecuciones, registro de errores y mantenimiento preventivo para detectar fallos a tiempo y aplicar correcciones oportunas.
- **Evolución y escalabilidad del pipeline (actualización + PySpark/ELT):** Dado que los sistemas conectados pueden cambiar (portal ACEPTA, interfaces internas o estructuras de datos), se recomienda ampliar la cobertura a más fuentes y mayores volúmenes de datos, se sugiere migrar progresivamente a tecnologías como PySpark en Python e incorporar un enfoque ELT (Extract, Load, Transform), habilitando análisis avanzado y escenarios de grandes datos.
- **Automatización de la corrección de inconsistencias:** Se recomienda como trabajo de investigación futuro ampliar el sistema para que, además de identificar discrepancias, aplique reglas de negocio que permitan corregir o normalizar inconsistencias de manera automática (y cuando no sea posible, generar propuestas de corrección y un flujo de revisión/aprobación), asegurando trazabilidad de los cambios.

- **Seguridad y control de accesos:** Reforzar políticas de seguridad y control de accesos (gestión de credenciales, roles y auditoría) debido a la sensibilidad de la información contable y financiera procesada por el sistema.

Bibliografía

Abolacio Bosch, M. (2013). *Gestión contable*. IC Editorial.

ACEPTA Perú. (s.f.). Soluciones electrónicas para empresas.

Aggarwal, C. C. (2015). *Data Mining: The Textbook*. Springer.

Amazon Web Services. (2020). AWS Cloud Adoption Framework.

<https://aws.amazon.com/whitepapers/>

Amazon Web Services. (2024). *Documentación oficial de AWS*.

<https://docs.aws.amazon.com/>

Amazon Web Services. (2025). *¿Qué es AWS? - Computación en la nube con Amazon Web Services*. <https://aws.amazon.com/es/what-is-aws/>

Ardila, R. (2011). *La investigación científica*. Siglo XXI.

Automation Anywhere. (2025a). *What is Attended vs Unattended Robotic Process Automation (RPA)?*

<https://www.automationanywhere.com/rpa/attended-vs-unattended-rpa>

Automation Anywhere. (2025b). *What is Robotic Process Automation (RPA)? An Enterprise Guide*.

<https://www.automationanywhere.com/rpa/robotic-process-automation>

Axmann, B., & Harmoko, H. (2022). Process & Software Selection for Robotic Process Automation (RPA). *Technical Journal*, 16(3), 412-419.

<https://doi.org/10.31803/tg-20220417182552>

BBVA Spark. (2023a). *Conciliación bancaria: qué es, cómo funciona y sus ventajas* [Consultado el 17 de agosto de 2025].

<https://www.bbvaspark.com/es/noticias/conciliacion-bancaria-que-es-ventajas/>

BIBLIOGRAFÍA

BBVA Spark. (2023b). *Gestión de tesorería de una empresa: ¿qué es y cómo gestionarla?* [Consultado el 17 de agosto de 2025].
<https://www.bbvaspark.com/es/noticias/gestion-tesoreria-empresa-que-como-gestionarla/>

Bhagya, H. (2023). *Automating Browser Interactions Using Selenium in Python*. Real Python. <https://realpython.com/modern-web-automation-python-selenium/>

Bisquerra, R. (2014). *Metodología de la investigación educativa* (3.^a ed.). Editorial La Muralla.

Cook, T. D., & Campbell, D. T. (2005). *Quasi-Experimentation: Design and Analysis Issues*. Houghton Mifflin.

Costello, T., & Blackshear, L. (2019). *Prepare Your Data for Tableau. A Practical Guide to the Tableau Data Prep Tool*. <https://doi.org/10.1007/978-1-4842-5497-4>

Deemer, P., Benefield, G. L., & Vodde, B. (2009). Información básica de Scrum (The scrum primer).

EY. (2025). *Gestión del cumplimiento fiscal* [Consultado el 17 de agosto de 2025].
https://www.ey.com/es_cl/services/tax/gestion-del-cumplimiento-fiscal

EY Perú. (2024). *Nuevos horizontes de la madurez digital en el Perú*.
https://www.ey.com/es_pe/insights/technology/nuevos-horizontes-de-la-madurez-digital-en-el-peru-2024

Gelinas, U. J., Dull, R. B., & Wheeler, P. R. (2018). *Accounting Information Systems* (11.^a ed.). Cengage Learning.

Hernandez, R., Fernandez, C., & Baptista, P. (2014). *Metodología de la investigación* (6.^a ed.). McGraw-Hill Education.

Hurtado-Guevara, R. F. (2024). Impacto de la Automatización Contable en la Eficiencia Operativa de las PYMEs. *Revista Científica Zambos*, 3(1), 19-35.
<https://doi.org/10.69484/rcz/v3/n1/10>

IBM. (2025). *What is Robotic Process Automation (RPA)?* [Accedido el 17 de abril de 2025]. <https://www.ibm.com/think/topics/rpa>

BIBLIOGRAFÍA

Lizana, L. R. (2018). *Automatización en la recolección, tratamiento y envío de información estadística médico-asistencial a la Superintendencia Nacional de Salud (SUSALUD) basado en procesos de ETL y RPA para la clínica Adventista Ana Stahl* [Tesis de licenciatura]. Universidad Peruana Unión, Facultad de Ingeniería y Arquitectura.

López, J. (2021). Gestión contable y financiera en el sector retail: Retos y tendencias. *Revista Peruana de Administración*, 45-57.

Machado, G. V., Cunha, I., Pereira, A. C., & Oliveira, L. B. (2019). DOD-ETL: distributed on-demand ETL for near real-time business intelligence. *Journal of Internet Services and Applications*, 10(21).
<https://jisajournal.springeropen.com/articles/10.1186/s13174-019-0121-z>.

Ochoa Surco, A., & Osorio Schuler, P. (2022). *Implementación de un Robotics Process Automation (RPA) para mejorar el proceso de validación de estados de cuenta en una entidad financiera*, Lima-2022.
<https://repositorio.utp.edu.pe/handle/20.500.12867/6478>

Oracle. (s.f.). ¿Qué es un sistema de gestión financiera?

Palacios Martel, M. J. (2022). *Sistema ETL para la mejora en el movimiento de información entre servidores no comunicados en el Data Warehouse MIS de Financiera Confianza S.A.A.*
<https://repositorio.uncp.edu.pe/handle/20.500.12894/7196>

Pandas Development Team. (2024). *Pandas: Python Data Analysis Library*.
<https://pandas.pydata.org/>

Pressman, R. S. (2014). *Ingeniería del software: un enfoque práctico* (7.^a ed.). McGraw-Hill.

Salas, B. N. (2023). *Sistema que automatice el proceso de cuadre diario de las transferencias interbancarias inmediatas de los bancos en el Perú* [Tesis de pregrado]. Universidad Peruana de Ciencias Aplicadas.
<http://hdl.handle.net/10757/671931>

BIBLIOGRAFÍA

SAP. (s.f.). *¿Qué es un sistema de gestión financiera?*

<https://www.sap.com/latinamerica/products/erp/s4hana/what-is-financial-management-system.html>

Schwaber, K., & Sutherland, J. (2020). La Guía Definitiva de Scrum: Las Reglas del Juego.

Sommerville, I. (2011). *Ingeniería del software* (9.ª ed.). Pearson.

SUNAT. (2021). *Uso de tecnologías emergentes en la administración tributaria del Perú.*

<https://www.sunat.gob.pe/cuentassunat/planestrategico/memoria/memoria2021.pdf>

SUNAT. (2023). Guía de infracciones y sanciones tributarias.

https://cultura.sunat.gob.pe/sites/default/files/Guia_infracciones-sanciones-tributarias_2023.pdf

Tamayo, M. (2011). *Metodología de la investigación científica*. Limusa.

The Selenium Project. (2023). *Selenium with Python*.

<https://selenium-python.readthedocs.io>

Topitop. (s.f.). Acerca de nosotros : TOPITOP.

Van Der Aalst, W. M., Bichler, M., & Heinzl, A. (2018). Robotic Process Automation.

En *Robotic Process Automation* (pp. 269-272).

https://doi.org/10.1007/978-3-319-66963-2_7

Varela, J. P., Canale, S., & De Ponti, H. (2018). Los Sistemas Contables de Gestión en Organizaciones. *XIII Jornadas de Investigación*. <https://fce.unl.edu.ar/jornadasdeinvestigacion/trabajos/uploads/trabajos/54.pdf>.

Wewerka, J., & Reichert, M. (2020). Robotic Process Automation – A Systematic

Literature Review and Assessment Framework. *arXiv preprint*.

<https://arxiv.org/abs/2012.11951>

Zhang, C. A., Issa, H., Rozario, A., & Sveistrup, S. J. (2023). Robotic Process Automation (RPA) Implementation Case Studies in Accounting: A Beginning to End Perspective. *Accounting Horizons*, 37(1).

<https://doi.org/10.2308/HORIZONS-2021-084>