



Extracción, transformación y carga de información de PQR's de Tigo

Cristian Camilo Rendón Jimenez

Informe de práctica empresarial para optar al título de ingeniero electrónico

Asesor interno

Juan Pablo Urrea Duque, PhD. Profesor universitario

Asesor externo

Gisselle Cárdenas, Analista de información

Universidad de Antioquia

Facultad de ingeniería

Ingeniería electrónica

Medellín

2023

Cita	(Rendón Jimenez, C., 2023)
Referencia	Rendón Jimenez, C. (2023). <i>Extracción, transformación y carga de información de PQR's de Tigo</i> , Semestre de industria. Universidad de Antioquia, Medellín, Colombia.
Estilo APA 7 (2020)	



Centro de Documentación Ingeniería (CENDOI)

Repositorio Institucional: <http://bibliotecadigital.udea.edu.co>

Universidad de Antioquia - www.udea.edu.co

Rector: John Jairo Arboleda Céspedes.

Decano/Director: Julio César Saldarriaga Molina.

Jefe departamento: Augusto Enrique Salazar Jiménez.

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

Dedicado a mis padres, mi pareja y a toda mi familia, por su amor, apoyo incondicional y paciencia en todo mi proceso de formación.

Agradecimientos

Agradecimientos a mis padres, mi pareja y a toda mi familia, por su apoyo y motivación constante. A la Universidad de Antioquia por permitirme formarme como profesional íntegro. Y a la compañía Tigo, en especial a la gerencia Analytics, por los aprendizajes brindados y darme la confianza para desarrollar este proyecto.

Tabla de contenido

Resumen	9
Abstract	10
Introducción	11
1 Objetivos	13
1.1 Objetivo general	13
1.2 Objetivos específicos	13
2 Marco teórico	15
2.1 Sistema gestor de bases de datos	15
2.1.1 Oracle Database	16
2.1.2 Microsoft SQL Server	16
2.1.3 SQL Server Integration Services (SSIS)	16
2.2 Proceso ETL	16
2.2.1 Extracción	16
2.2.2 Transformación	16
2.2.3 Carga	17
2.3 Python	17
2.3.1 Pyodbc	17
2.3.2 Cx_Oracle	17
2.3.3 Pandas	18
2.4 Tableau	18
2.5 KPI	18
3 Metodología	19
3.1 Onboarding	20

3.2 Desarrollo de la ETL	21
3.2.1 ETL para PQR's de Une y Tigo	22
3.2.2 ETL para PQR's de Edatel	23
3.2.3 Proceso automático de la ETL	26
3.3 Limpieza y homologación	26
3.4 Liquidación de factura	29
3.5 KPI de reincidencia	30
3.6 Dashboard	32
3.6.1 Preparación de los datos	32
3.6.2 Construcción del tablero en Tableau	34
3.7 Carga automática de información detallada	34
4 Resultados	37
4.1 Tablas actualizadas en servidor local	37
4.2 Apoyo al proceso para liquidación de factura	38
4.3 Construcción del KPI de reincidencia	38
4.4 Dashboard de PQR's general	38
4.5 Proceso automático de carga del detalle de PQR's	44
6 Conclusiones	45
Referencias	47

Lista de tablas

Tabla 1 Ejemplo de tabla de mapeo para homologación de campos

28

Lista de figuras

Figura 1 Etapas del proyecto	19
Figura 2 Diagrama detallado del proyecto	20
Figura 3 Conexiones en Oracle	21
Figura 4 Bloques ETL en SSIS	22
Figura 5 Componente Data Flow Task de SSIS	23
Figura 6 Componente Execute Process Task de SSIS	24
Figura 7 Proceso bat para ETL Edatel	24
Figura 8 Diagrama de flujo para ETL Edatel	25
Figura 9 Proceso bat para ETL completa	26
Figura 10 Consultas SQL para liquidación de factura	30
Figura 11 Consultas SQL para cálculo de reincidencia	32
Figura 12 Consultas SQL para fuentes de datos en Tableau	33
Figura 13 Proceso automático para carga del detalle	35
Figura 14 Programación de tarea para carga automática del detalle	36
Figura 15 Tablas de PQR's creadas en el servidor ReportingLeader	37
Figura 16 Dashboard PQR's - Ingresos	39
Figura 17 Dashboard PQR's - Cierres	41
Figura 18 Dashboard PQR's – Reincidencia por mes	42
Figura 19 Dashboard PQR's – Reincidencia por grupo	42
Figura 20 Dashboard PQR's – Empoderamiento	43
Figura 21 Ejecución de la tarea automática de carga del detalle de PQR's	44
Figura 22 Ruta y carpetas creadas para el acceso al detalle de PQR's	44

Siglas, acrónimos y abreviaturas

ETL	Extract, Transform and Load
SSIS	SQL Server Integration Services
KPI	Key Performance Indicator
SQL	Structured Query Language
PQR's	Peticiones, Quejas y Reclamos
BU	Business Unit
UE	Unidad estratégica
B2B	Business to Business
B2C	Business to Customer
APC	Atención en primer contacto

Resumen

El presente proyecto se enfocó en la extracción y carga de información de Peticiones, Quejas y Reclamos (PQR's) de Tigo y sus filiales (UNE, TIGO, EDATEL), para su posterior procesamiento, visualización y análisis. Inicialmente, se identificó que el proveedor encargado de atender las PQR's compartía la información diariamente en un repositorio a partir de archivos de Excel sin la posibilidad de actualizar el estado de los casos radicados, lo que dificultaba su seguimiento y gobernanza. Por lo tanto, se propuso la extracción de datos dispuestos por el proveedor y su almacenamiento en un servidor propio de la compañía con SQL Server, lo que permitiría su actualización y acceso mediante una única conexión y autenticación, así como la generación de un histórico consultable.

Para la extracción de datos de Tigo y UNE se utilizó la herramienta SSIS, mientras que para Edatel se empleó Python debido a incompatibilidades con el gestor de base de datos Oracle. Posteriormente, se realizó la limpieza y transformación de los datos mediante consultas SQL para su entrega al área encargada de tener control del posible cobro de las PQR's atendidas del mes. Asimismo, se diseñó y se implementó un KPI sobre la reincidencia de las PQR's, que se automatizó en SQL para su obtención diaria.

Por último, se llevó a cabo el despliegue de un dashboard para la visualización de información de las PQR's e indicadores relacionados, permitiendo la toma de decisiones de negocio. En conclusión, el trabajo logró la automatización del proceso de extracción y carga de datos de PQR's, su procesamiento mediante SQL y la visualización de resultados mediante el dashboard, lo que contribuye a la mejora de la gobernanza de los datos, a la toma de decisiones y a la optimización de los procesos de la compañía Tigo y sus filiales.

Palabras clave: PQR's, ETL, extracción de datos, limpieza y transformación de datos, dashboard, gobernanza de datos, KPI.

Abstract

This project focused on the extraction and loading of Petitions, Complaints and Claims (PQRs) information from Tigo and its subsidiaries (UNE, TIGO, EDATEL) for subsequent processing, visualization, and analysis. Initially, it was identified that the provider responsible for managing the PQRs shared information daily in a repository from Excel files without the possibility of updating the status of filed cases, which made it difficult to track and govern. Therefore, it was proposed to extract the data provided by the provider and store it in a company-owned server with SQL Server, allowing for updating and access through a single connection and authentication, as well as the generation of a searchable history.

For the extraction of data from Tigo and UNE, the SSIS tool was used, while Python was used for Edatel due to incompatibilities with the Oracle database management system. Subsequently, data cleaning and transformation were performed through SQL queries for delivery to the area responsible for controlling possible charges for the attended PQRs of the month. Likewise, a KPI on the recurrence of PQRs was designed and implemented, which was automated in SQL for daily obtaining.

Finally, a dashboard was deployed to display information on PQRs and related indicators, enabling business decision-making. In conclusion, the work achieved the automation of the extraction and loading process of PQR data, its processing through SQL, and the visualization of results through the dashboard, contributing to the improvement of data governance, decision-making, and process optimization of Tigo and its subsidiaries.

Keywords: PQRs, ETL, data extraction, data cleaning and transformation, dashboard, data governance, KPI.

Introducción

La compañía Tigo está constituida actualmente por tres filiales que son, UNE: para servicios fijos de televisión, internet y telefonía, Tigo: para servicios de telecomunicaciones móviles y EDATEL: para servicios fijos de internet y telefonía. Anteriormente, se obtenía la información de peticiones, quejas y reclamos (PQR's) a partir de archivos de Excel (fuente antigua) que el proveedor, encargado de atender este tipo de requerimientos de los usuarios, compartía diariamente en un repositorio con información del día anterior, sin la posibilidad de actualizar el estado de los casos radicados. Como parte de sus actividades, el proveedor dispuso la información de PQR's de TIGO, UNE y EDATEL en servidores externos con diferentes conexiones cada uno, a partir del cual se podría consultar la data a través de SQL con el gestor de base de datos Oracle, tercerizando la responsabilidad de la información.

No obstante, es necesaria la extracción de los datos dispuestos por el proveedor, para guardarla en un servidor propio de la compañía con SQL Server, buscando que la información sea accesible por tener una única conexión y autenticación, sumado a la necesidad de obtener un histórico que pueda ser consultado en una ventana de tiempo determinada, permitiendo que el área tenga mayor gobernanza sobre estos datos en los diferentes procesos que lo requieran, así como la toma de decisiones estratégicas. Uno de los procesos a impactar es el de facturación del proveedor, porque permite obtener la información de las PQR's atendidas por dicho proveedor y de esta manera se puede hacer seguimiento a los detalles de esta factura al final del mes.

La extracción de datos sobre PQR's de las fuentes de UNE y Tigo, fue desarrollada a través de un proceso con la herramienta SSIS (SQL Server Integration Services), para así poder agregar diariamente la información en una tabla para Tigo y otra para UNE, en el principal servidor usado por el área. En cuanto a la extracción de datos de Edatel, se presentaron inconvenientes de compatibilidad entre la herramienta SSIS y el gestor de base de datos Oracle, por tanto, se plantea abarcar la solución del problema realizando el proceso ETL por medio de Python a través de lotes de registros, de manera que se adicione en serie al proceso de carga de datos de las demás filiales y se garantice completitud de los datos.

Una vez realizado el proceso automático para la extracción y carga de los datos de cada fuente, se requiere limpiarlos y transformarlos con consultas SQL según determinados criterios ya establecidos contractualmente entre TIGO y el proveedor, y por consiguiente entregar el insumo (detalle y conteo por tipología) de PQR's atendidas al área encargada, para que así puedan tener control del posible cobro (facturación) que hará el proveedor de las PQR's atendidas y cerradas del mes.

Acto seguido, se realiza el diseño de un KPI sobre la reincidencia de las PQR's y su respectiva implementación en SQL para obtenerlo de manera automática. Por otra parte, se realiza un proceso automático en Python para la carga mensual en una ruta en la red corporativa, y así los interesados puedan tener la información detallada de los casos reincidentes, facturados y del dashboard. Finalmente, se realiza el despliegue de un dashboard sobre PQR's para la generación de conocimiento a través de visualizaciones, ayudando al análisis de toda la información relacionada y la toma de decisiones de negocio.

1 Objetivos

1.1 Objetivo general

Extraer de fuentes foráneas la información de PQR's de TIGO y sus filiales (Une, Tigo, Edatel) y cargarla en un servidor local garantizando consistencia, exactitud e integridad de los datos, para apoyar procesos de liquidación del proveedor, así como la generación y visualización de indicadores clave de rendimiento (KPIs) para futuros análisis cualitativos y cuantitativos en la vicepresidencia de Experiencia a clientes.

1.2 Objetivos específicos

- Extraer automáticamente la información de PQR's de las filiales Une y Tigo a través de SSIS, y de Edatel utilizando python por medio de lotes de registros, y cargarla de manera diaria en un servidor local para unificarla en un repositorio con los datos de PQR's de las filiales Tigo y Une.
- Limpiar y homologar datos de PQR's Tigo con el lenguaje de consulta SQL, eliminando registros repetidos, unificando tipologías redundantes, modificando el formato de fechas y eliminando caracteres especiales no necesarios.
- Transformar, filtrar y obtener a través de consultas SQL, 44 campos necesarios para control de las PQR's atendidas por el proveedor y que son facturadas por este, basados en lineamientos y políticas ya establecidas para factura entre el proveedor y Tigo con sus respectivas filiales, como el documento de identidad, fechas de apertura y cierre, tipo de PQR, número de radicado, entre otros.
- Construir y obtener KPIs relacionados con PQR's tales como volumen por tipo de PQR, tiempos de resolución, reincidencia de casos de PQR's, rata de cierre de tickets por día y por mes y empoderamiento de los canales de atención, para su posterior análisis y toma de decisiones por parte de la vicepresidencia de experiencia a clientes.

- Visualizar la información sobre KPIs de PQR's Tigo a través de dashboard en Tableau para el consumo y seguimiento de los indicadores por parte de las áreas de la empresa interesadas en PQR's y así, puedan tomar decisiones de negocio basadas en los datos.

2 Marco teórico

Los datos se han convertido en un activo esencial para las empresas para comprender y responder a las necesidades cambiantes de los clientes, tomar decisiones informadas y obtener una ventaja competitiva en el mundo actual impulsado por los datos. En la industria de las telecomunicaciones, la experiencia del cliente se ha convertido en un factor crítico para el éxito empresarial, para lo cual es necesario comprender, a través de los datos, las inconformidades, necesidades y preferencias de los clientes para ofrecer productos y servicios personalizados y mejorar los niveles de satisfacción de los usuarios.

Particularmente, las peticiones, quejas y reclamos (PQR's), son una fuente valiosa de información, ya que brindan una retroalimentación directa de los usuarios acerca de los problemas que enfrentan al usar los productos o servicios de la empresa. Por lo tanto, es fundamental que las compañías tengan sistemas eficaces de gestión y análisis de datos para recopilar información de los diferentes canales por los cuales ingresan las PQR's y poder tomar medidas adecuadas para solucionar las quejas de los usuarios y mejorar la experiencia del cliente en general.

En este contexto, se hace necesario contar con herramientas que permitan manejar grandes cantidades de datos, que a su vez permitan realizar procesos de extracción, transformación y carga de datos, así como herramientas de visualización para facilitar el análisis y la toma de decisiones. A continuación, se presentan términos cruciales para el desarrollo del proyecto, relacionados con la gestión y análisis de datos.

2.1 Sistema gestor de bases de datos

Un sistema gestor de base de datos o DBMS (*Data Base Management System*), es un programa de software que permite administrar los datos que ya se encuentran de manera estructurada, ya sea configurando o extrayendo la información que se encuentra en la base por medio de interfaz gráfica y/o con el lenguaje de consulta respectivo, siendo el más popular SQL (*Structured Query Language*).

2.1.1 Oracle Database

Oracle Database es el gestor de bases de datos comercial más antiguo, lanzado en 1979. Este gestor permite crear estructuras de bases de datos por medio de SQL y su lenguaje propio PL/SQL que incorpora funcionalidades adicionales para la gestión de los datos (IONOS, 2020).

2.1.2 Microsoft SQL Server

Es uno de los gestores más utilizados en aplicaciones de business intelligence (Pérez, 2021), el cual utiliza el lenguaje de desarrollo propio TSQL (*Transact-SQL*) y la interfaz gráfica SQL Server Management Studio (*SSMS*) para el manejo y administración de los componentes de las bases de datos.

2.1.3 SQL Server Integration Services (SSIS)

Es un componente de Microsoft SQL Server que permite migrar, transformar e integrar datos de orígenes de datos relacionales, archivos planos o XML.

2.2 Proceso ETL

Un proceso de extracción, transformación y carga (ETL: *Extract, transform and Load*), consiste en un conjunto de procedimientos para garantizar el flujo de datos entre diferentes sistemas de almacenamiento de información, que puede ser desde múltiples fuentes hacia un único almacén de datos. Entre las principales aplicaciones, está la creación de un Data Warehouse que permite unificar los datos de diferentes bases en un solo almacén para análisis de información en las empresas, la integración en sistemas heredados para migrar datos y actualizar aplicaciones existentes (Espinosa, 2009), y la consolidación de datos para sincronización de sistemas.

2.2.1 Extracción

Recolectar datos necesarios a partir de diferentes fuentes de información.

2.2.2 Transformación

Operar sobre los datos extraídos como puede ser filtrar, limpiar, depurar, seleccionar campos específicos y obtener nuevos campos de interés.

2.2.3 Carga

Depositar la información en un almacén de datos donde quedan listos para poder ser analizados.

2.3 Python

Es un lenguaje de programación de alto nivel, interpretado, interactivo y orientado a objetos, utilizado en diversas áreas como lo es la ingeniería de datos, incluyendo la construcción de ETLs y la conexión a bases de datos, gracias a su facilidad de uso y la gran cantidad de librerías especializadas disponibles.

2.3.1 Pyodbc

Es un módulo de Python que proporciona una interfaz para acceder a bases de datos a través de ODBC. Se utiliza para conectarse a fuentes de datos, realizar extracciones y cargar datos en un destino. Pyodbc es compatible con diferentes tipos de bases de datos, lo que lo hace muy útil en entornos donde se necesita trabajar con múltiples fuentes de datos.

2.3.1.1 ODBC

Open Database Connectivity, es una interfaz de programación de aplicaciones (API) para acceder a bases de datos, se trata de una tecnología que permite la conexión y acceso a diferentes bases de datos desde una aplicación utilizando un conjunto común de comandos. Fue desarrollada por Microsoft en los años 90 y se ha convertido en un estándar para la conexión y acceso a bases de datos relacionales.

2.3.2 Cx_Oracle

Es un módulo de Python que permite la conexión y acceso a bases de datos Oracle desde aplicaciones Python. Es una biblioteca que proporciona una interfaz de programación de aplicaciones (API) para conectarse a una base de datos Oracle, ejecutar comandos SQL y recuperar resultados en Python.

2.3.3 Pandas

Es una biblioteca de Python que se utiliza principalmente para el análisis y la manipulación de datos. Ofrece estructuras de datos flexibles y de alto rendimiento, diseñadas específicamente para trabajar con datos relacionales y etiquetados, lo que lo hace muy útil en tareas como la limpieza, filtrado, agrupación y transformación de datos.

2.4 Tableau

Es una herramienta de visualización de datos interactiva que permite a los usuarios conectar y visualizar datos de múltiples fuentes, creando gráficos, dashboards y otros tipos de representaciones visuales de los datos. Al convertir los análisis en representaciones visuales, permite consumir la información de manera más fácil, lo cual es de suma importancia porque la visualización de datos es una herramienta esencial en las organizaciones para comunicar información compleja de manera clara y concisa, lo que puede ayudar a tomar decisiones más informadas (Wexler et al., 2017).

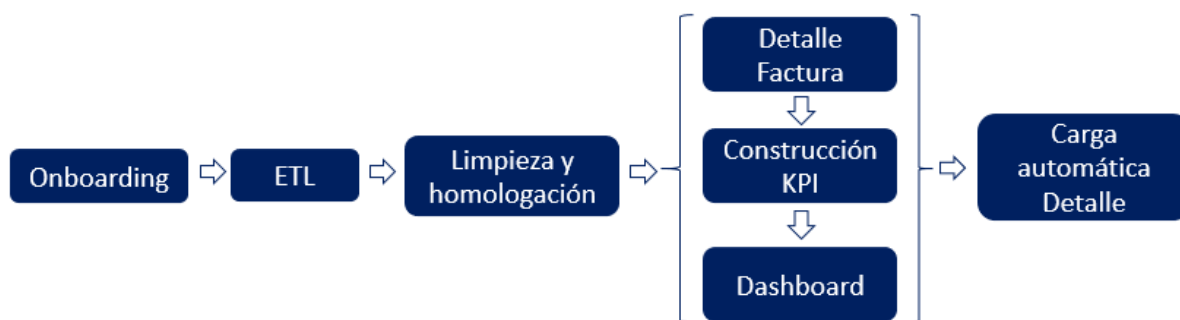
2.5 KPI

Un indicador clave de rendimiento (Key performance indicator) permite efectuar la medición de una meta u objetivo con el fin de evaluar las acciones estratégicas de negocio que se toman y alinear al equipo de trabajo para su cumplimiento. Los KPIs deben estar alineados con los objetivos estratégicos de la empresa y medir el progreso hacia esos objetivos de una manera clara y significativa (Kenny, 2020). La fórmula de cálculo para medir el KPI debe ser clara y sencilla de entender, y debe reflejar la relación entre el resultado que se desea medir y las variables que lo afectan. Para ello es necesario obtener datos relevantes que pueden provenir de diferentes fuentes, como bases de datos, sistemas de gestión, encuestas, entre otros.

3 Metodología

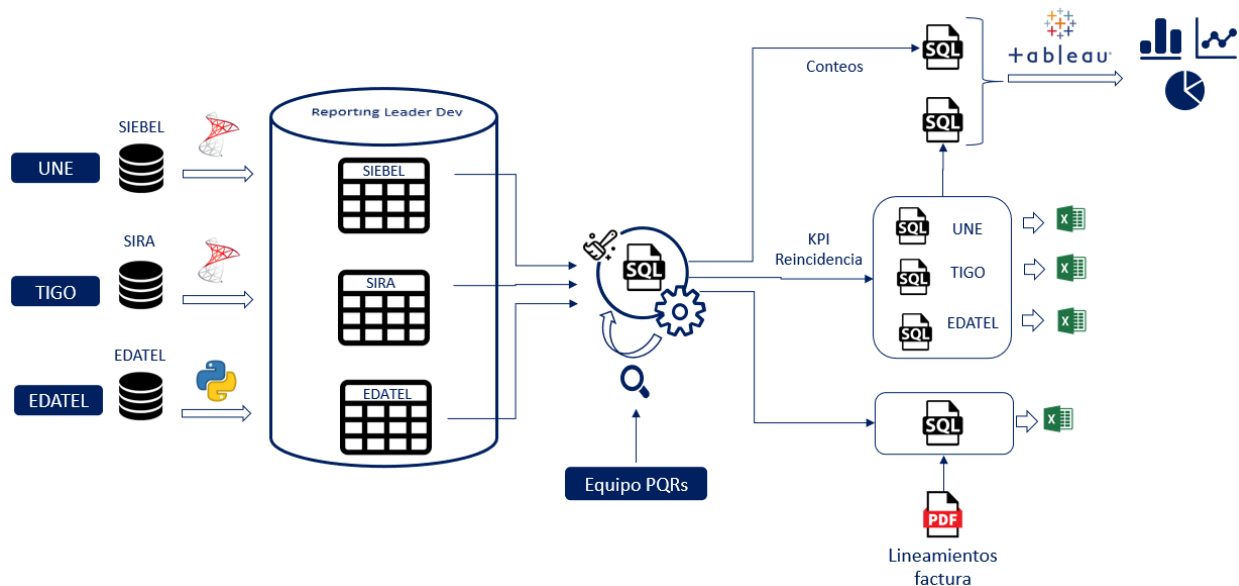
El proyecto se ha llevado a cabo a través de siete etapas clave, tal como se muestra en la figura 1. En la primera etapa, denominada Onboarding, se ha destinado al entendimiento del negocio y los requerimientos del proyecto. Luego, se ha procedido con la implementación de la ETL y la limpieza y homologación de datos, con el objetivo de depurar los datos y garantizar su calidad para su posterior uso en las tres etapas siguientes. Estas etapas incluyen la obtención del detalle para la liquidación de factura, la construcción del KPI de reincidencia y el despliegue del dashboard. Finalmente, se encuentra la etapa de carga automática del detalle a partir de lo desarrollado en las tres etapas anteriores.

Figura 1
Etapas del proyecto



Cada una de estas etapas se describe en detalle a continuación, donde se explicará cómo se han llevado a cabo, así como las herramientas y técnicas que se han utilizado para su implementación. La figura 2 muestra un diagrama más detallado de las etapas, incluyendo información técnica adicional, desde la extracción de datos desde cada una de las fuentes para cada filial, la carga en las tablas respectivas en el servidor local, el tratamiento realizado a través de SQL para homologar y limpiar los datos con ayuda de las observaciones y entendimiento del negocio del equipo de PQR's, hasta el desarrollo del KPI, del dashboard y el proceso de liquidación de factura.

Figura 2
Diagrama detallado del proyecto



3.1 Onboarding

Esta etapa se destina al entendimiento del negocio, la comprensión de los antecedentes y contextualización de la situación inicial sobre la información de PQR's. Inicialmente, se comprende la conformación de la empresa TIGO por parte de las filiales Une, Tigo y Edatel, por lo que la radicación de PQR's se debe hacer de manera aislada para cada razón social y, por ende, se cuenta con tres diferentes fuentes donde queda alojada la información de PQR's de cada filial.

La fuente antigua de PQR's se basaba en archivos de Excel que el proveedor, encargado de atender los casos, disponía en una ruta con los casos gestionados diariamente. Esta manera de obtener la información hacía que la fuente fuera estática al no poder tener una actualización del estado de los casos ingresados, además de haber una dependencia operativa adicional por parte del proveedor que a su vez pudiese ocasionar errores humanos. Por tanto, se planteó la obtención de la información de PQR's directamente desde las bases de datos donde se encuentra alojada la información de los usuarios incluyendo las radicaciones de PQR's realizadas por el proveedor, que si bien estas bases son propiedad de TIGO, era necesario unificar la información en una única base de datos de manera local a la vicepresidencia de experiencia a clientes, para así no tener diferentes autenticaciones para acceder a los datos de cada filial y poder acceder de una manera ágil y rápida,

debido a que las bases de datos se encuentran en Oracle y sus conexiones son sumamente ralentizadas.

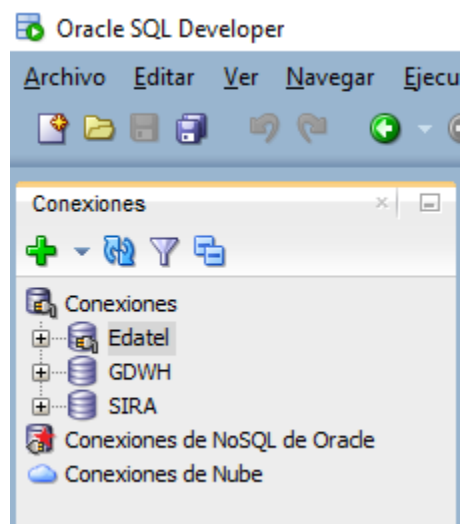
Como se indica en la figura 2, la información de la filial Une se encuentra en una base de datos llamada SIEBEL, para la filial Tigo se encuentra en una base de datos con el nombre de SIRA y para la filial Edatel en otra base de datos con el mismo nombre. Las tres bases de datos con el gestor de base de datos Oracle, a partir de las cuales se inicia el proceso de extracción de datos.

3.2 Desarrollo de la ETL

El propósito de esta etapa es obtener tres tablas en el servidor local (llamado “*Reporting Leader Dev*”) con la información de PQR’s actualizada diariamente para cada filial. La extracción y carga de las tablas para las filiales Une y Tigo, mostradas en la figura 2 como “SIEBEL” y “SIRA” respectivamente, fue realizada por medio del software SSIS, mientras que para la filial Edatel fue realizada en Python.

Primeramente, se obtienen las autenticaciones para las conexiones de Oracle y así acceder a las bases de datos donde se aloja la información de cada filial. El proveedor comparte las tres consultas SQL para la obtención de los campos necesarios.

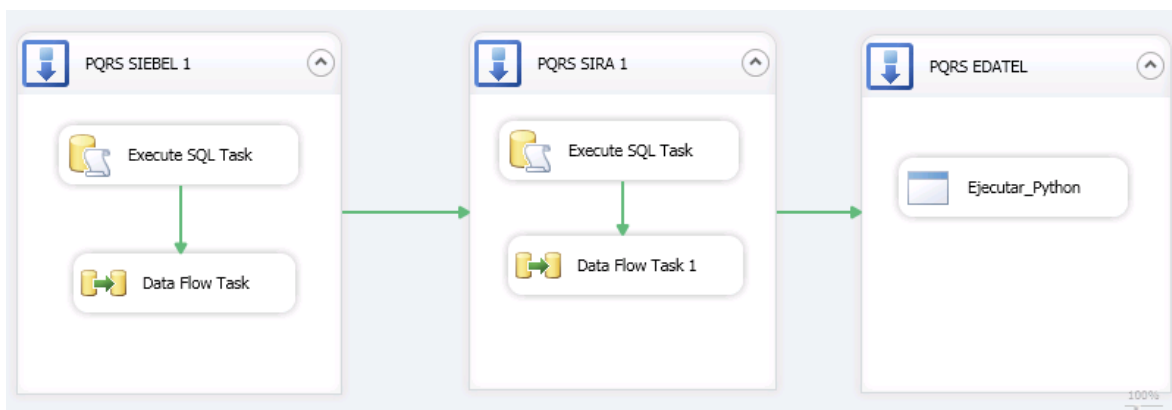
Figura 3
Conexiones en Oracle



Una vez verificadas las conexiones en Oracle y los resultados de las consultas SQL, se modifican para agregar en un proceso en SSIS y un proceso en Python para Edatel, con el fin de primero obtener todos los casos ingresados desde el mes de junio 2022 y así tener el histórico hasta la fecha inicial en que se realiza el proceso. Es importante aclarar que luego de tener el histórico, se realiza la automatización para que cada día se elimine la información de la tabla en el servidor local de los últimos 30 días y se agreguen los 30 días más recientes (esto para cada una de las tres tablas). Con lo anterior, se busca mantener actualizada la información que se almacena en cada tabla, asegurando que, si una PQR ha cambiado de estado de un día a otro, por ejemplo, de abierto a anulado o de abierto a cerrado, se actualice en las tablas locales.

En la siguiente figura se muestran los bloques en SSIS implementados para la ETL de cada filial, los cuales se explican a detalle a continuación.

Figura 4
Bloques ETL en SSIS



3.2.1 ETL para PQR's de Une y Tigo

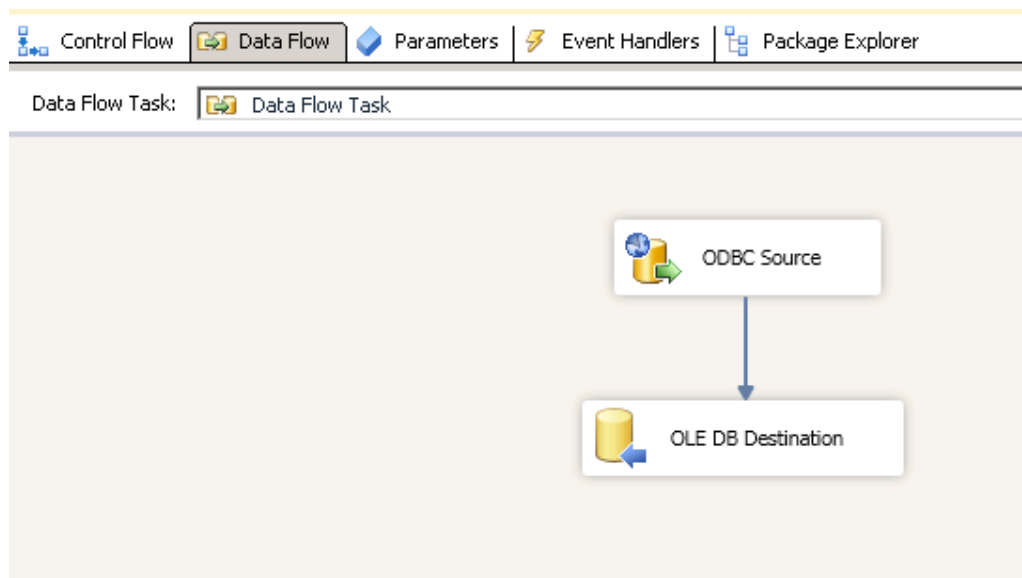
En la figura 4, el bloque “PQRS SIEBEL 1” corresponde al proceso de extracción y carga de las PQR's de la filial Une, mientras que el bloque “PQRS SIRA 1” corresponde a la filial Tigo. Ambos bloques son similares, y se diferencian en las consultas que se ejecutan para obtener los datos desde Oracle y las tablas a las que apuntan en SQL Server en el servidor local.

El componente “Execute SQL Task”, es utilizado para eliminar los últimos 30 días de la tabla, tanto para Une como para Tigo, para lo cual se ejecuta una instrucción DELETE en el

lenguaje SQL. El componente “Data Flow Task” es utilizado para consultar por medio de SQL los 30 días más recientes en la base de datos Oracle y cargarlos en la tabla de la base de datos en SQL Server. En este elemento, se agrega internamente el componente “ODBC Source” por medio del cual se realiza la conexión a Oracle y se ejecuta la consulta SQL, conectado al componente “OLE DB Destination” que contiene la conexión a SQL Server apuntando a la tabla en la cual se desean cargar los datos. En el caso de Une, la tabla en la cual se cargan los datos se nombró “TBL_PQRS_SIEBEL_COMPLETAS” y para Tigo se nombró “TBL_PQRS_SIRA_COMPLETAS”.

Figura 5

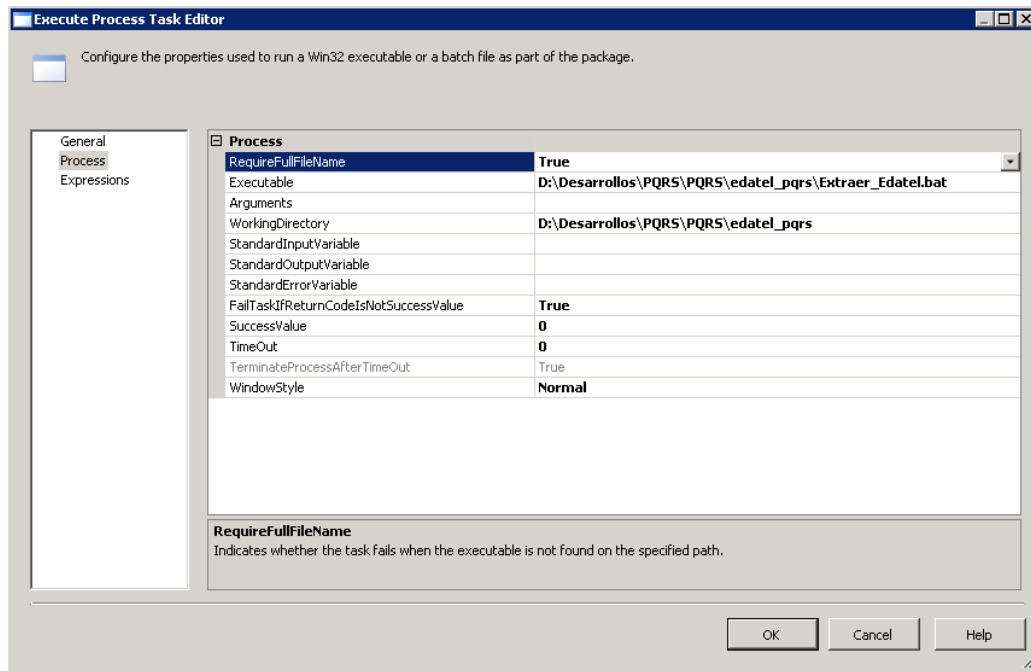
Componente Data Flow Task de SSIS



3.2.2 ETL para PQR's de Edatel

Debido a incompatibilidad entre SSIS y la versión de Oracle de la base de datos en la cual se encuentra la información de PQR's de Edatel, se optó por desarrollar la ETL en Python a través de lotes de registros. Si bien en la figura 4, se evidencia que hay un bloque llamado “PQRS EDATEL”, este solo se encarga de ejecutar un proceso .bat, el cual ejecuta la ETL en Python, como se muestra en el componente “Execute Process Task” de la figura 6, donde el archivo ejecutable es “Extraer_Edatel.bat”.

Figura 6
Componente Execute Process Task de SSIS



Las instrucciones del ejecutable “Extraer_Edatel.bat” se muestran en la figura 7. El único objetivo es activar el entorno virtual de Python para ejecutar el archivo main, encargado de realizar el proceso de extracción y carga de datos de Edatel.

Figura 7
Proceso bat para ETL Edatel

```
@echo off

:: cambiar de directorio
cd D:\
cd D:\Desarrollos\PQRS\PQRS\edatel_pqrs

:: activar el entorno virtual
call venv\Scripts\activate.bat

:: limpiar consola
cls

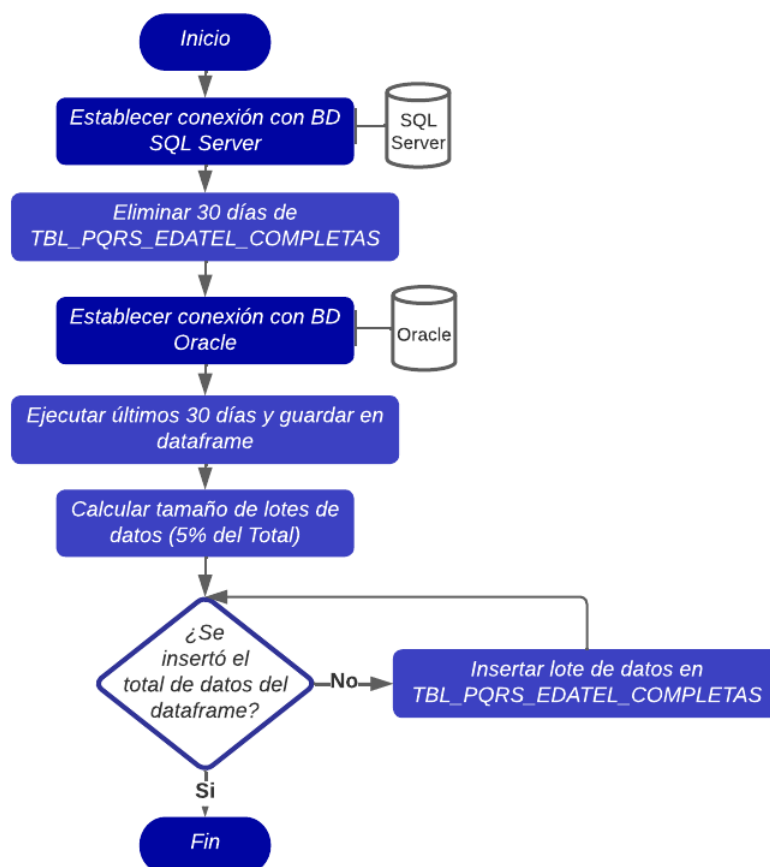
:: Ejecutar main
py main.py

:: desactivar el entorno virtual
call venv\Scripts\deactivate.bat
```


El proyecto en Python, utiliza la librería *cx_oracle* para conectarse a la base de datos Edatel en Oracle y la librería *Pyodbc* para conectarse a la base de datos SQL Server. Primeramente, se ejecuta la consulta “eliminar.sql” para eliminar los 30 días de la tabla “TBL_PQRS_EDATEL_COMPLETAS”. Luego se ejecuta la consulta “extraer.sql” con Oracle, que arroja los registros de las PQR’s en los últimos 30 días y se almacena el resultado en un dataframe de Pandas. Finalmente, se insertan los datos en la tabla de la base de datos SQL Server, dividiendo los datos en pequeños lotes correspondientes al 5% del total de registros para mejorar el rendimiento. En la figura 8, se muestra el diagrama de flujo que ilustra el proceso realizado para la ETL de Edatel.

Figura 8

Diagrama de flujo para ETL Edatel



3.2.3 Proceso automático de la ETL

Una vez se tiene completo el proyecto en SSIS con los componentes que se ilustraron en la figura 4, y se han realizado pruebas de funcionamiento para que la carga de datos de cada filial se realice de manera adecuada, se procede a automatizar la ETL, para lo cual se crea un archivo .bat (figura 9) para que se ejecute el archivo Package.dtsx, con el cual se ejecuta todo el proyecto SSIS. Finalmente, se crea una nueva tarea en el programador de tareas de Windows para que el archivo “Tarea_CargaR_PQRS.bat” se ejecute de manera diaria. Todo esto desarrollado en el servidor de aplicaciones del área de Analytics – vicepresidencia de experiencia a clientes.

Figura 9

Proceso bat para ETL completa

```
D: > Desarrollos > PQRS > PQRS > Tarea_CargaR_PQRS.bat
1 C:
2 CD C:\Program Files (x86)\Microsoft SQL Server\110\DTS\Binn
3 DTExec.exe /FILE "D:\Desarrollos\PQRS\PQRS\PQRS\Package.dtsx" /CHECKPOINTING OFF /REPORTING EWCDI
4
5 PAUSE
6 EXIT|
```

3.3 Limpieza y homologación

Una vez que se tienen las tres tablas en el servidor local al área, en la base de datos SQL Server “Reporting Leader Dev”, se inicia un proceso de comunicación con el equipo de PQR’s, incluyendo reuniones e intercambio de correos electrónicos con el fin de definir las convenciones y los requerimientos necesarios para efectuar los procesos de liquidación de factura, la construcción del KPI de reincidencia y del dashboard.

Se realiza una exploración en SQL de los campos que contiene cada tabla de PQR’s, la cantidad de registros ingresados y cerrados, diarios y mensuales, agrupados por tipo de PQR, por el estado (abierto, cerrado o anulado), el medio de ingreso (Tiendas, medios digitales, contact center), la favorabilidad de la PQR (a favor o en contra del cliente), el grupo que atiende la PQR (si es atención en primer contacto – APC, o si es escalada al grupo de atención Back), entre otros. Lo anterior considerando que en cada tabla los campos son nombrados de manera diferente, así pues, se podría encontrar que para Une el campo de favorabilidad se llama “RESULTADO”, para Tigo “TIPO_RESPUESTA” y para Edatel “ESTADO_RECLAMO”. Esto implica un reto en

cuanto a la homologación de campos, dado que al tratarse de PQR's abarcan un componente legal que implica el manejo de particularidades por razón social, por lo que para homologar cada campo fue necesaria la comunicación con cada persona encargada de cada filial. Esta etapa abarca un trabajo arduo de constante retroalimentación por parte del equipo de PQR's y de esta manera hacer las respectivas correcciones con base en los resultados de la etapa del dashboard y del KPI reincidencia.

La metodología para llevar a cabo las homologaciones con el equipo de PQR's fue a través de tablas de mapeo, lo cual consiste en obtener para cada campo de interés las diferentes combinaciones para que el experto en materia de PQR's pudiera elegir el valor correspondiente. De esta manera, al equipo de PQR's se le compartía una tabla en Excel por cada campo. Por ejemplo, para la homologación del medio de ingreso en el caso de Une, se compartió la información de la tabla 1, con los campos FUENTE y ESTADO, y la persona encargada devolvía el archivo con el medio de ingreso homologado correspondiente. También, es importante acotar que se realizaron convenciones de manera que los medios de ingreso quedaran nombrados igual para las tres filiales, lo cual permite que todas las PQR's que ingresaron por medios digitales, sin importar la filial, quedaran nombradas como "Ecare". De manera similar, se procede con los demás campos.

Luego de tener las tablas de mapeo completas, se empiezan a construir consultas estructuradas en SQL para así trasladar la información de las homologaciones a condiciones en el lenguaje SQL de dos maneras. La primera es aplicable en los casos en que, para obtener el campo homologado, no se necesita un gran número de combinaciones, como es el caso del ejemplo ilustrado en la Tabla 1, el cual es posible traducir a SQL a través de sentencias *CASE WHEN*. La segunda forma de trasladar las homologaciones, se aplica cuando las combinaciones son una cantidad considerablemente alta, para lo cual es necesario crear una tabla en la base de datos, cargar la tabla de mapeos del campo y hacer el cruce (sentencia *JOIN*) con la tabla de Une, Tigo o Edatel. Por ejemplo, para obtener el grupo de atención en Une, se requiere la validación de cuatro campos o columnas, para poder indicar en la quinta columna del grupo homologado si el grupo de atención fue APC o fue BACK. El total de combinaciones de las cuatro columnas es de 1610 por lo que en este caso se crea una tabla en la base de datos para cruzar.

Tabla 1*Ejemplo de tabla de mapeo para homologación de campos*

FUENTE	ESTADO	MEDIO_INGRESO_HOMOLOGADO
Gestión Proactiva	-	Back
Back	-	Back
Medios sociales	-	Ecare
WhatsApp	-	Ecare
Redes Sociales	-	Ecare
Mi Tigo	-	Ecare
Correo Interno	-	Medios Escritos
Carta	-	Medios Escritos
NULL	Abierto	Sin fuente-Abierto
NULL	Anulado	Sin fuente-Anulado
IVR	-	Teléfono
Telefono RPA	-	Teléfono
Teléfono	-	Teléfono
Presencial	-	Tiendas
Web-Online	-	Web
SMS WEB	-	Web
Web	-	Web

En la etapa de limpieza y homologación, se crean en total tres tablas en la base de datos para cruzar con las tablas de cada filial, nombradas:

- TBL_PQRS_GRUPOS_UNE
- TBL_PQRS_CAUSAS_CATEGORIAS
- DIAS_LABORALES

La primera tabla para obtener el grupo de atención para la filial Une, la segunda para homologar las causas y subcausas hacia categorías y subcategorías que agruparan los motivos por los cuales se radicaron las PQR's y la tercera, para calcular la edad de la PQR, la cual indica la duración desde que ingresó la PQR hasta ser resuelta, excluyendo los días no hábiles.

Otra de las actividades cruciales para las etapas siguientes, es la limpieza de los campos de descripción y solución, que son campos de texto libre en los cuales el asesor que atiende la PQR

y/o el cliente, colocan la descripción del problema y la solución que se le brinda. Para ello se eliminan caracteres especiales reemplazándolos con SQL por caracteres nulos.

Por último, se utiliza la tabla nombrada TBL_pymes_Base_ValoracionClientes, la cual ya venía siendo usada por el área para la segmentación de los clientes. Esta segmentación permite identificar a través del documento o NIT, cuales clientes son empresariales (segmento B2B) y cuales son personas naturales (segmento B2C).

3.4 Liquidación de factura

El proceso de facturación consiste en la obtención de los registros de PQR's necesarios para liquidación de la factura de los casos que el proveedor atiende mensualmente, ya que la empresa Tigo paga al proveedor por transacción realizada y cada valor de transacción varía de acuerdo a un conjunto de reglas contractualmente ya establecidas a través de un manual de lineamientos para extracción y verificación de factura. Este manual tiene un capítulo para cada filial, dado que la facturación se rige de distintas maneras para cada una.

La verificación se venía realizando por el equipo de PQR's manualmente por medio de Excel para comparar las PQR's correspondientes de pago con sus respectivas marcas y clasificaciones, con las PQR's que el proveedor indica haber atendido mensualmente. Por tanto, el equipo de PQR's solicitó al área de Analytics desarrollar una manera de hacer la verificación automática y así poder agilizar el proceso de extracción y verificación de factura. Para ello, era necesario primeramente tener la fuente de datos oficial para lo cual se desarrolló la ETL de los datos de PQR's y así poder iniciar a realizar los filtros partiendo de las bases de datos del total de casos atendidos del mes.

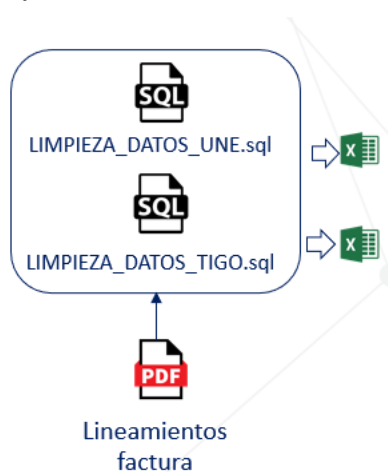
Dado que el manual está diseñado para filtrar y crear nuevas columnas en Excel, se parte traduciendo las instrucciones a realizar en Excel a un equivalente en SQL. Se decide realizar el proceso por medio de tablas temporales o CTEs, con sentencias *WITH AS*, lo que permite ir verificando en cada paso los filtros o acciones realizadas, y así continuar con el paso siguiente. Por ejemplo, en la primera tabla temporal se seleccionan todas las PQRs cerradas en el mes de análisis (mes en el cual se desea verificar la factura), se agrega un campo llamado PERIODO convirtiendo

la fecha de cierre al formato año-mes, y un campo más nombrado `CONDICION_PROPIETARIO` para saber si el asesor que creó la PQR fue el mismo que la cerró. Luego de verificar el resultado, se puede utilizar en la siguiente subconsulta hasta llegar al total de registros con todos los filtros aplicados y las columnas adicionales necesarias.

Al final, se construyen entonces dos consultas SQL con los filtros indicados en el manual, una para Une y otra para Tigo, cuyos resultados se exportan a Excel para compartir con el equipo de PQR's.

Figura 10

Consultas SQL para liquidación de factura



3.5 KPI de reincidencia

Las PQR's constituyen una parte esencial del proceso de atención al cliente de una empresa, sin embargo, cuando los clientes tienen que presentar múltiples PQR's sobre el mismo problema, puede indicar que hay una falta de resolución efectiva, un inconveniente recurrente que necesita ser abordado o una brecha en la atención al cliente. Para abordar esta problemática, se hace necesario la creación de un indicador que permita monitorear qué tanto los clientes se están contactando para colocar PQR's reiteradamente. Con esto, se busca mejorar la satisfacción del cliente, identificar patrones de comportamiento, evaluar la eficacia de las acciones tomadas por parte de los asesores de la operación y encontrar oportunidades de mejora en los procesos y servicios ofrecidos.

Dicho indicador es llamado Reincidencia, el cual mide la cantidad de PQR's cerradas del mes que se convierten en reincidentes porque un mismo usuario ha realizado una PQR anterior en un periodo de 90 días, sobre la cantidad total de PQR's cerradas en el mes. Su cálculo se muestra a continuación:

$$\%Reincidencia = \frac{\text{Cantidad de casos reincidentes del mes (en 90 días)}}{\text{Total de casos cerrados del mes}}$$

Para la obtención de este indicador se hace uso de SQL, realizando dos subconsultas, una con la cantidad de casos cerrados en el mes de análisis y otra con la cantidad de casos cerrados en cualquier periodo, para luego unirlos verificando que correspondan al mismo usuario a través de la cédula o NIT, que sean iguales en el segmento (B2B o B2C), el tipo de PQR (Petición, Queja factura, Queja servicio, Recurso, etc.), pero que el número de caso o ticket sea diferente, para así asegurar que se trata de una nueva PQR. Adicionalmente, se verifica que la fecha de cierre de una PQR en la subconsulta que contiene casos cerrados en cualquier periodo, esté dentro de un periodo de 90 días antes de la fecha de creación de otra PQR en la subconsulta que contiene las PQR's cerradas en el mes de análisis.

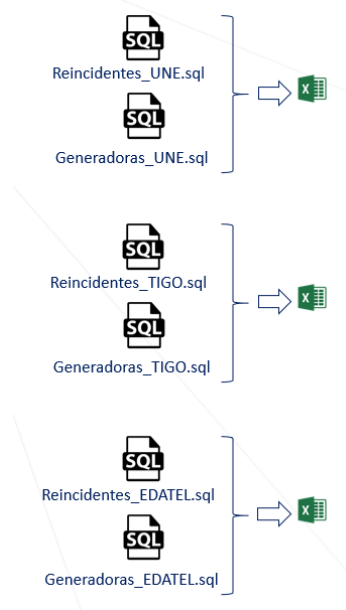
En síntesis, para saber cuántas PQR's reincidentes ha presentado alguien en un mes, se necesita revisar todas las PQR's que ha presentado esa persona o empresa en cualquier momento (no solo en ese mes) y luego compararlas con las PQR's que esa misma persona o empresa ha presentado en ese mes en particular. Para asegurarse de que son PQR's diferentes, se compara su número y su fecha de cierre, y se asegura de que no haya más de 90 días entre el cierre de una PQR y la creación de la siguiente. Si se encuentra una PQR cerrada en el mes que cumple con estos criterios, se considera reincidente y se suma al numerador.

Con lo anterior, es posible obtener también el detalle y la cantidad de PQR's generadoras, que corresponden a las PQR's que preceden a las reincidentes del mes. Es decir, si un usuario presenta una PQR por un problema con un producto y luego vuelve a presentar otra PQR dentro de 90 días por el mismo problema, podemos decir que la primera PQR es la PQR generadora de la

PQR recurrente. Esto permite identificar de una manera más clara la razón por lo que un usuario se recontacta e imputar al grupo que atendió la primera PQR y al canal por medio del cual ingresó.

Esta misma lógica en SQL, se aplica para las tres filiales, con la excepción de que, para la filial Tigo, al tratarse de solo telefonía móvil, se compara el número de teléfono en vez de el número de documento para identificar si la PQR es o no recurrente. Dado esto, se construyen entonces 6 consultas SQL que retornan el detalle de los casos para ser exportados a Excel, como se muestra en la figura 11, 3 consultas correspondientes a los casos recurrentes del mes para cada filial y 3 consultas correspondientes a las PQR's generadoras de reincidencia en cada filial.

Figura 11
Consultas SQL para cálculo de reincidencia



3.6 Dashboard

Esta etapa consta de dos partes principales, que son la preparación de los datos en SQL y la construcción del tablero en Tableau.

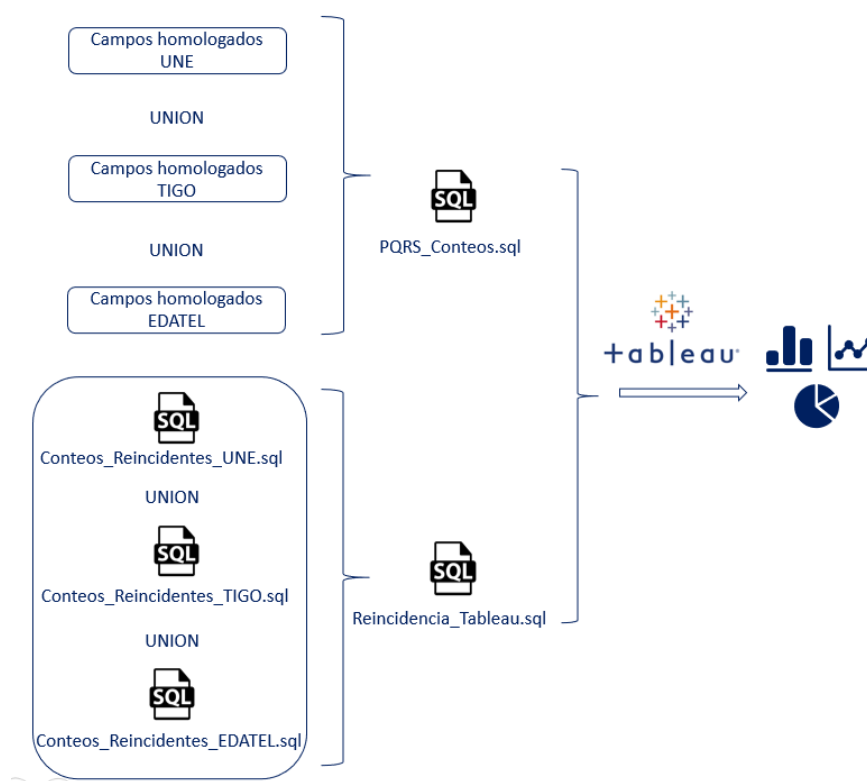
3.6.1 Preparación de los datos

Esta sección implica la identificación y corrección de errores en los datos, la combinación de diferentes fuentes de datos y la transformación de los datos en una forma que sea más fácil de

analizar y visualizar. Antes de poder visualizar los datos en un tablero, es necesario asegurarse de que los datos estén limpios y estructurados, lo cual se llevó a cabo principalmente a partir de lo desarrollado en la etapa de homologación y limpieza, y con las consultas de la etapa del KPI de reincidencia.

Los datos usados en el dashboard se prepararon a partir de dos consultas, como se muestra en la figura 12, una para realizar los conteos de ingresos y de cierres de PQR's (*PQRS_CONTEOS.sql*), y otra para realizar los conteos de casos reincidentes y así lograr visualizar el KPI de reincidencia (*Reincidencia_Tableau.sql*).

Figura 12
Consultas SQL para fuentes de datos en Tableau



La consulta de *PQRS_CONTEOS* se obtuvo a partir de la unión de los campos homologados de cada filial, luego de esto se agrupan los datos por fecha de ingreso, fecha de cierre, tipo de PQR, categoría, subcategoría, favorabilidad, medio de ingreso, canal de ingreso, grupo de atención, estado de la PQR, unidad de negocio BU (si es servicios fijos o móviles), filial, edad de la PQR,

los campos corte de apertura y corte de cierre, y se realiza el conteo de los distintos números de radicado de PQR. Con esto, se logra obtener el conteo de PQR's por las distintas combinaciones de los campos mencionados para los últimos 8 meses, lo cual garantiza que el tablero sea liviano y rápido al momento de interactuar con él, ya que, en vez de cargar todos los campos con un promedio de 2 millones de registros, solo se cargan 17 campos con alrededor de 390.000 registros.

En cuanto a la consulta *Reincidencia_Tableau*, se procede de manera similar, solo que primeramente se realiza el agrupamiento de campos en cada filial, partiendo de las consultas desarrolladas en la etapa del KPI de reincidencia y luego se lleva a cabo la unión de los campos agrupados con el respectivo conteo de la cantidad de generadoras, la cantidad de PQR's cerradas en el mes y la cantidad de PQR's cerradas con reincidencia a 90 días.

3.6.2 Construcción del tablero en Tableau

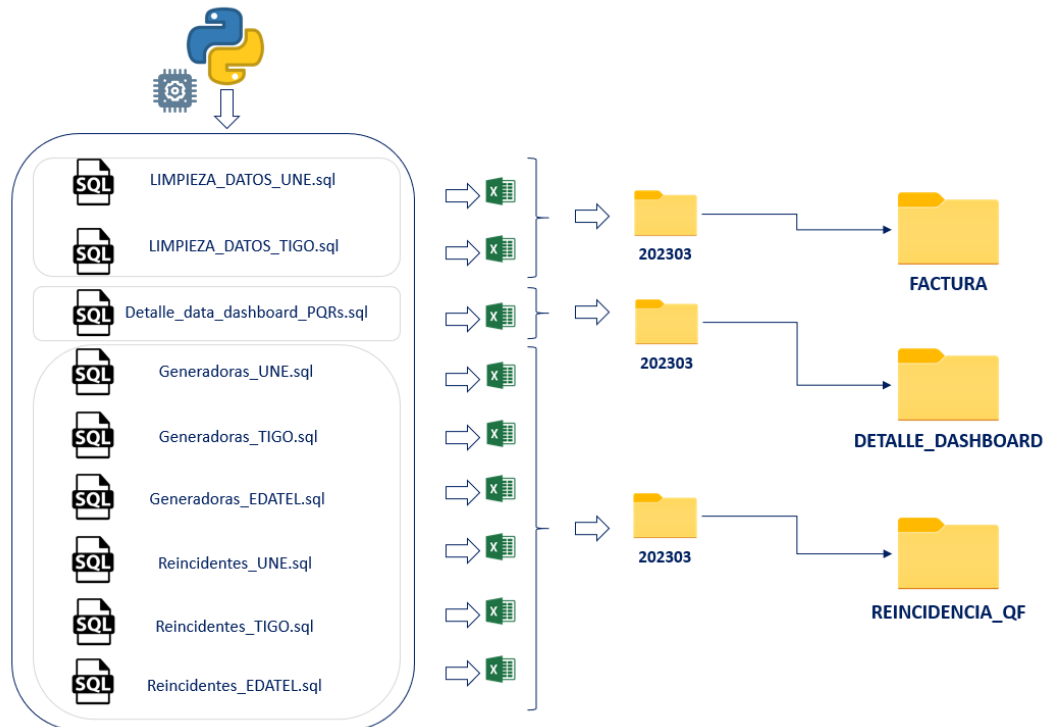
La construcción del tablero en Tableau inicia desde la selección de las fuentes de datos a usar, que en este caso corresponden a las consultas SQL que se prepararon en la sección anterior. Se crean las gráficas y tablas en hojas separadas. En cada hoja, se selecciona la fuente de datos. Luego, se puede dar formato a la visualización, agregar etiquetas y títulos, y ajustar el tamaño y la ubicación de la visualización en la hoja. Una vez que se han construido todas las hojas necesarias, se combinan en un tablero para crear una vista global a través de 4 páginas, correspondientes a ingresos, cierres, reincidencia y empoderamiento. En el proceso de combinación, se puede ajustar el tamaño y la posición de cada hoja en el tablero.

3.7 Carga automática de información detallada

Con el fin de evitar manualidades y no tener la necesidad de ejecutar mensualmente las consultas SQL para luego exportar el detalle a archivos de Excel que luego se comparten con el equipo de PQR's, se realizó una automatización para que el primer día de cada mes se realice el exporte de archivos CSV en una ruta pública a la cual el equipo de PQR's pudiese acceder para revisar y analizar los archivos de verificación de factura, el detalle desglosado de los registros graficados en el dashboard y los archivos de reincidentes y generadoras para cada filial. El proceso se desarrolló en Python como se ilustra en la figura 13, se guardan los archivos en una carpeta

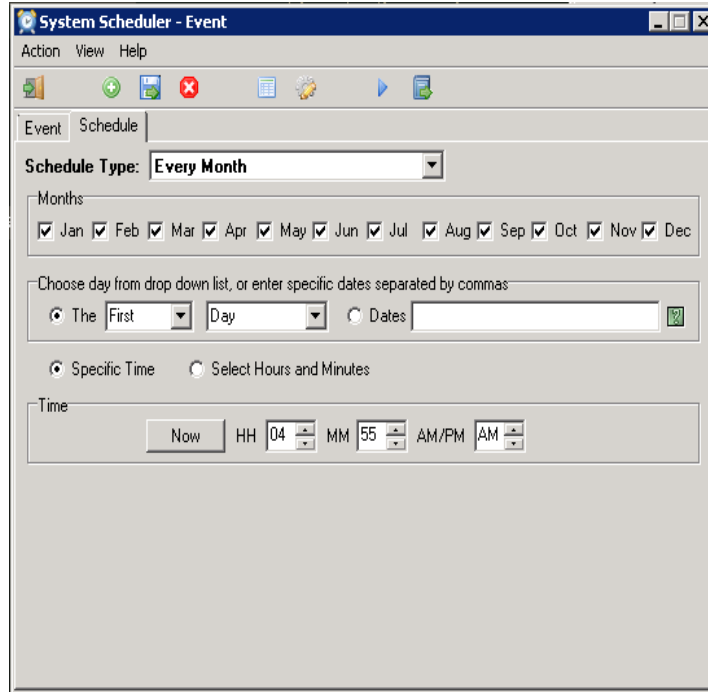
correspondiente al periodo dentro de cada una de las carpetas de factura, detalle del dashboard y reincidencia.

Figura 13
Proceso automático para carga del detalle



De manera análoga al desarrollo de la ETL de Edatel, el proceso en Python consiste en la conexión a la base de datos en SQL Server, solo que ahora todas las consultas corren en la misma de base de datos y en vez de cargarse la información en otra base de datos, se exportan a archivos CSV/Excel que se guardan en la ruta correspondiente. Se crea la tarea en el programador de tareas de Windows para que se ejecute el día uno de todos los meses a las 4:55am, como se evidencia en la figura 14.

Figura 14
Programación de tarea para carga automática del detalle



4 Resultados

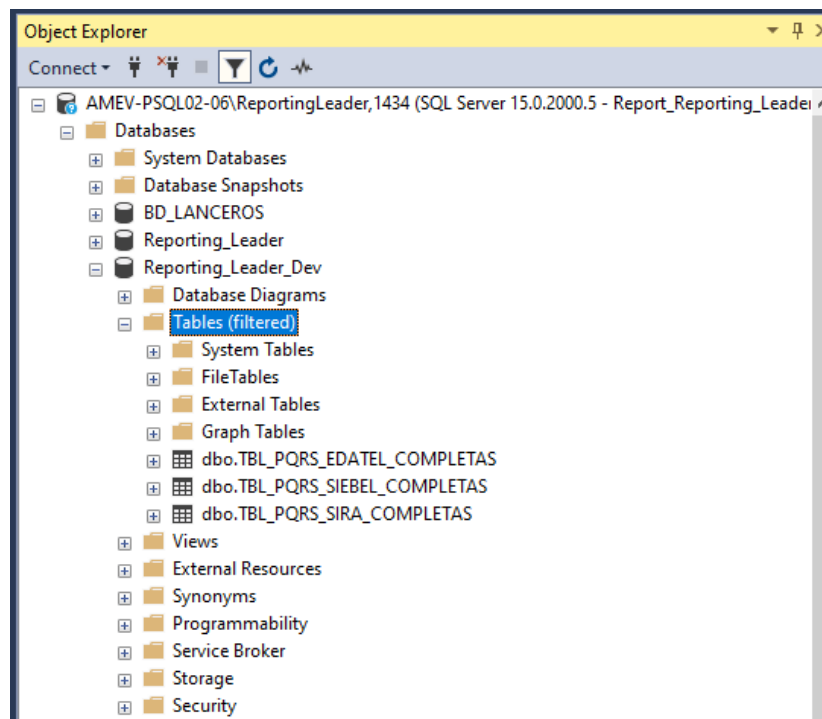
Al finalizar el proyecto, se logran obtener las tablas actualizadas diariamente de PQR's en el servidor local, un dashboard en Tableau para análisis de la información de PQR's, la construcción del KPI de reincidencia accesible tanto desde el tablero como desde el detalle en archivos de Excel y el proceso automático de carga de la data detalle para el proceso de facturación, el análisis minucioso del indicador de reincidencia y de la información del tablero.

4.1 Tablas actualizadas en servidor local

Como se ilustra en la figura 15, se obtuvieron las tres tablas de cada filial en el servidor local en la base de datos Reporting_Leader_Dev.

Figura 15

Tablas de PQR's creadas en el servidor ReportingLeader



4.2 Apoyo al proceso para liquidación de factura

Se obtienen las consultas SQL correspondientes para la verificación de factura para las filiales Une y Tigo, que arrojan para cada periodo de análisis los registros filtrados con las columnas adicionales según el manual de lineamientos de factura, a partir de las cuales el equipo de PQR's puede acordar el pago de factura con el proveedor que atiende los casos.

4.3 Construcción del KPI de reincidencia

Se logra la construcción del KPI de reincidencia para cada filial, el cual, como se verá en la siguiente sección, puede ser visualizado y monitoreado a través del tablero como también puede ser analizado a través del detalle de PQR's reincidentes y generadoras de reincidencia.

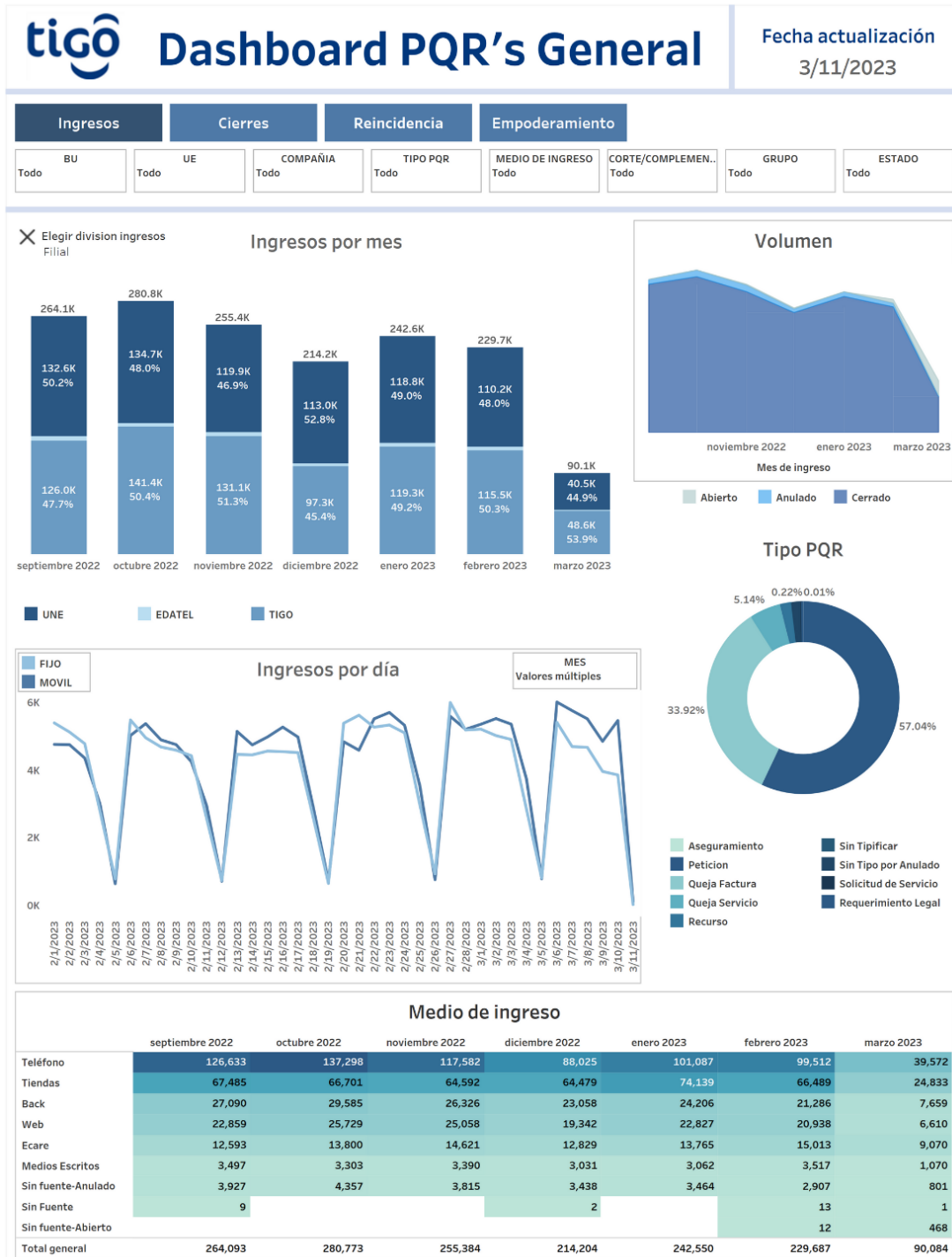
4.4 Dashboard de PQR's general

Se obtiene un dashboard en tableau que permite el análisis de la cantidad de ingresos, cantidad de cierres, del KPI de reincidencia y de empoderamiento. El dashboard se muestra desde la figura 16 a la figura 20.

En la figura 16 se muestra la pestaña de ingresos de PQR's, allí se visualiza con un gráfico de barras la cantidad de ingresos por mes con la posibilidad de ver el evolutivo del porcentaje que representa cada filial respecto al total del mes o el porcentaje del tipo de PQR respecto al total del mes. Se visualiza un gráfico de “volumen”, que permite visualizar la comparación entre los casos cerrados contra los abiertos y los anulados para cada mes. Se tiene un gráfico de líneas para ver el evolutivo por día de qué tantas PQR's ingresan diariamente, un gráfico de dona para ver la cantidad y el porcentaje por tipo de PQR, y finalmente, una tabla que muestra la cantidad de PQR's que ingresan por cada medio en cada mes.

Todo esto con la capacidad de interactuar con los filtros que se agregaron de unidad de negocio – BU (Fijo o Móvil), la unidad estratégica – UE (B2B o B2C), la compañía o filial (Une, Tigo, Edatel), el tipo de PQR (Petición, Queja Factura, Queja Servicio, Recurso, Requerimiento legal, etc.), el medio de ingreso (Teléfono, Tiendas, Web, Ecare, etc.), la opción corte o complemento (para ver toda la información hasta la fecha de corte), el grupo de atención (APC o BACK) y el estado (abierto, cerrado o anulado).

Figura 16
Dashboard PQR's - Ingresos



En la figura 17, se muestra la pestaña de cierres que contiene gráficas similares a la pestaña de ingresos, con gráficas de cierres por mes, cierres por día y tipo, pero se añade además un gráfico de dona de doble nivel para visualizar las causas por las cuales se radicaron las PQR's englobadas en categorías y subcategorías (facturación, soporte técnico, cancelación, información, etc.) y un gráfico de la edad de la PQR para visualizar cuántas PQR's se resolvieron en determinados grupos de días hábiles, considerando que el tiempo reglamentario para dar respuesta a una PQR puesta por un usuario es de 15 días hábiles. En esta pestaña se agregan los filtros de favorabilidad, para saber si una PQR resuelta fue a favor o en contra del cliente, y un filtro para seleccionar una categoría específica.

En la figura 18 y 19, se muestran los elementos de la pestaña de Reincidencia. En la figura 18, se puede ver un gráfico de líneas con el evolutivo mensual del KPI de reincidencia con la posibilidad de filtrar por el tipo de PQR, la filial y la unidad estratégica (B2B o B2C). Este gráfico tiene la opción de ver como líneas o como tabla, para así también poder ver el numerador (cantidad de casos reincidentes del mes) y denominador (total de casos cerrados del mes).

Se cuenta también con un botón "Detalle" en la parte superior derecha que redirige a la ruta en la red donde los usuarios del dashboard pueden consumir los archivos CSV/Excel, para analizar a detalle las PQR's reincidentes y generadoras.

En la parte inferior de la pestaña de reincidencia, se encuentra una tabla de reincidencia por grupo, como se muestra en la figura 19. Esta tabla funciona también con los filtros de la parte superior pero adicionalmente con el filtro del periodo específico. Con ella se puede identificar, por ejemplo, que 520 PQR's de tipo Queja factura, cerradas en el periodo 202302, fueron generadas por el grupo de atención APC y el medio de ingreso Teléfono, con 2 generadoras o casos anteriores. Lo cual ayuda a identificar la imputabilidad de reincidencia, es decir, asignar un grupo y medio de ingreso como responsable de que en el mes de febrero hubiera una gran cantidad de PQR's reincidentes, debido a que en la primera y segunda PQR puesta por el usuario (2 generadoras) no se le dio una solución definitiva, análisis que se puede complementar y verificar con la información detalle.

Figura 17
Dashboard PQR's - Cierres

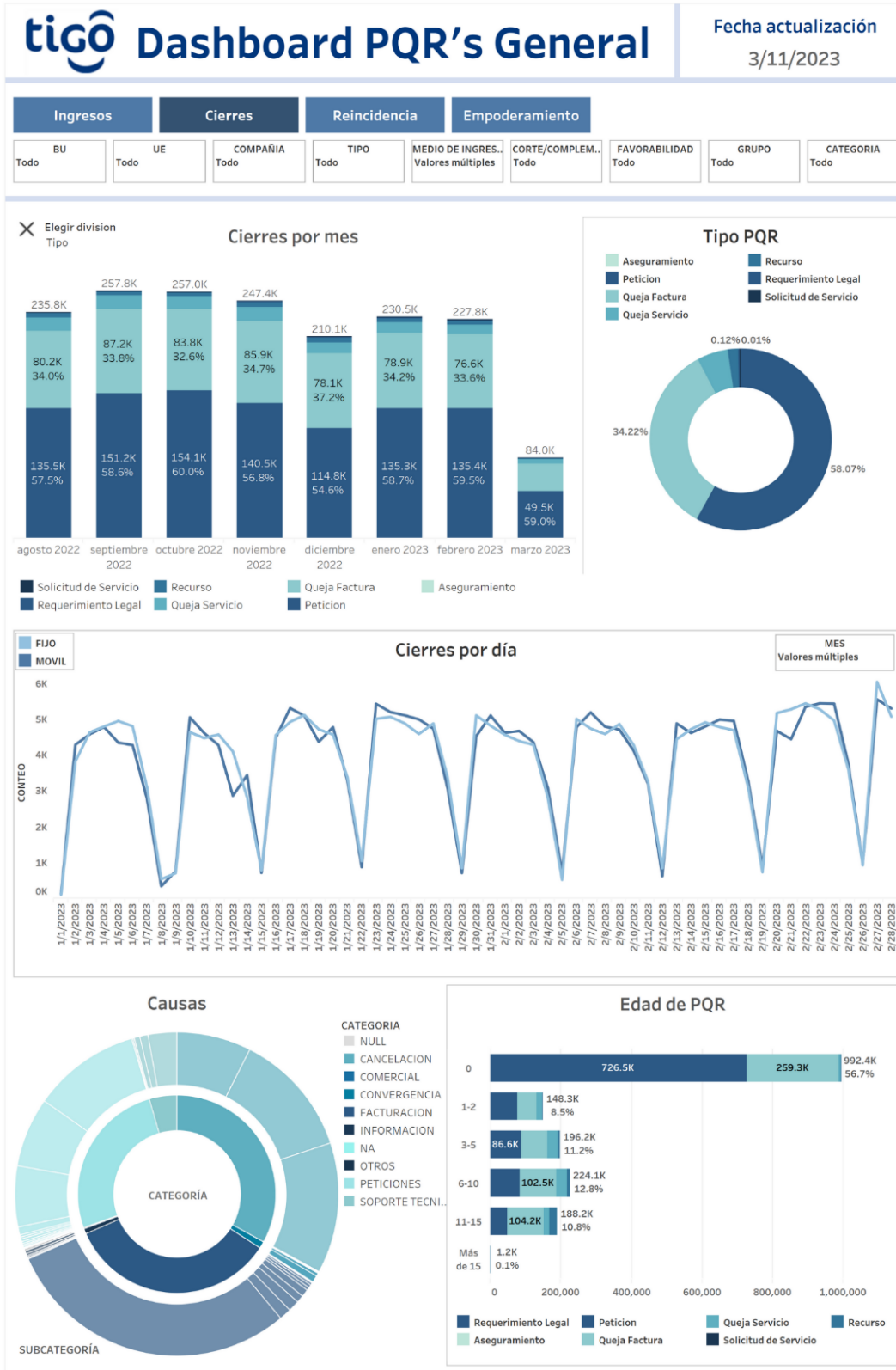


Figura 18
Dashboard PQR's – Reincidencia por mes

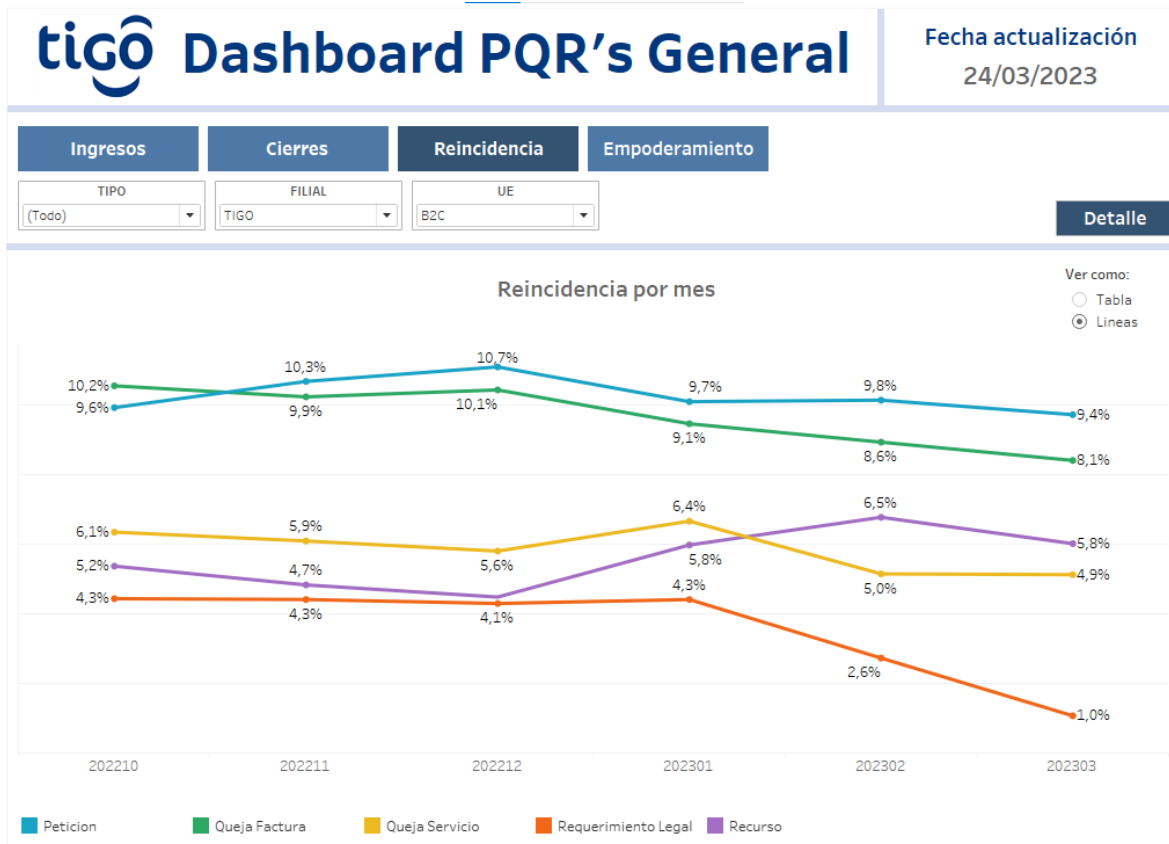


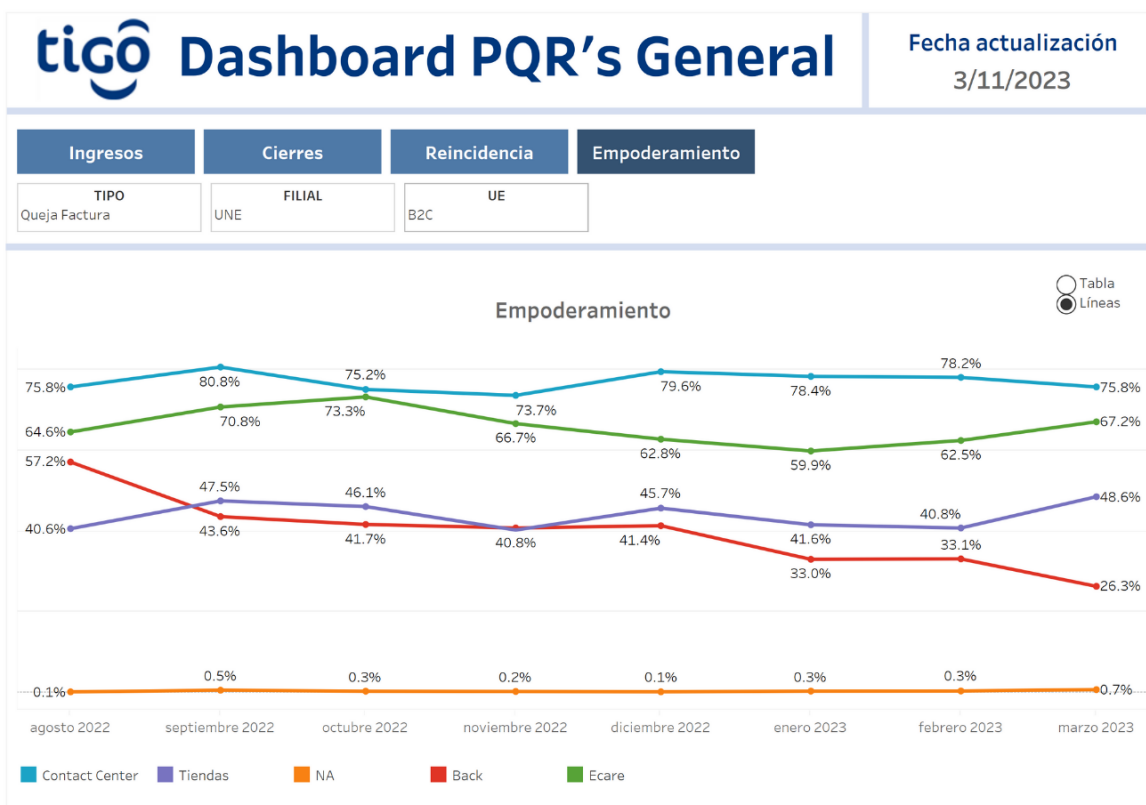
Figura 19
Dashboard PQR's – Reincidencia por grupo

PERIODO: 202302

REINCIDENTE	GRUPO_GEN	MEDIO_INGRESO_GEN	Cantidad de generadoras								
			1		2		+3		Total general		
			Total	%	Total	%	Total	%	Total	%	
SI	APC	Back	1	0,0%			9	0,0%		1	0,0%
		Digital	383	0,4%	48	0,0%		440	0,4%		
		Teléfono	4.399	4,2%	520	0,5%	98	0,1%	5.017	4,8%	
		Tiendas	2.228	2,1%	389	0,4%	204	0,2%	2.821	2,7%	
		Web	173	0,2%	27	0,0%	20	0,0%	220	0,2%	
		Total	7.184	6,8%	984	0,9%	331	0,3%	8.499	8,1%	
SI	BACK	Back	45	0,0%	9	0,0%	2	0,0%	56	0,1%	
		Digital	103	0,1%	14	0,0%	2	0,0%	119	0,1%	
		Teléfono	503	0,5%	70	0,1%	30	0,0%	603	0,6%	
		Tiendas	342	0,3%	43	0,0%	20	0,0%	405	0,4%	
		Web	154	0,1%	35	0,0%	29	0,0%	218	0,2%	
		Total	1.147	1,1%	171	0,2%	83	0,1%	1.401	1,3%	
Total	8.331	7,9%	1.155	1,1%	414	0,4%	9.900	9,4%			

Por último, se tiene la pestaña de empoderamiento, en la cual se encuentra un gráfico de líneas con el evolutivo del KPI de empoderamiento. Este KPI ya venía siendo utilizado por el área de PQR's, sin embargo, a través del dashboard se hizo disponible diariamente para que así pudieran acceder en cualquier día del mes y ver su comportamiento, sin necesidad de esperar a reportes a fin de mes, para obtenerlo de manera operativa por medio de Excel. El indicador es relevante analizarlo por canal de ingreso y permite identificar para cada canal, el porcentaje de PQR's que son atendidas en primer contacto (APC) sin necesidad de ser escaladas a otro grupo de atención especializada (Back). Por ejemplo, de las PQR's que ingresaron por Contact Center, cuántas fueron atendidas en primer contacto versus el total, lo cual da una idea de la autonomía y capacidad del canal para atender las PQR's sin necesidad de escalarlas a otro grupo.

Figura 20
Dashboard PQR's – Empoderamiento

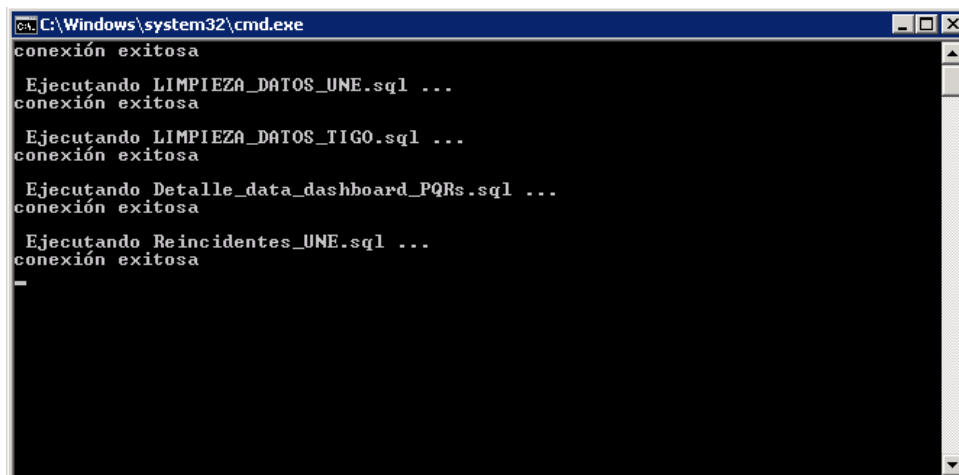


4.5 Proceso automático de carga del detalle de PQR's

Se logra obtener un proceso automático para evitar carga operativa de ejecutar consultas SQL y hacer exportes en Excel que luego son compartidos al equipo de PQR's. En la figura 21, se puede ver el proceso ejecutándose para cada una de las 9 consultas SQL, cuyos resultados son exportados en las rutas mostradas en la figura 22.

Figura 21

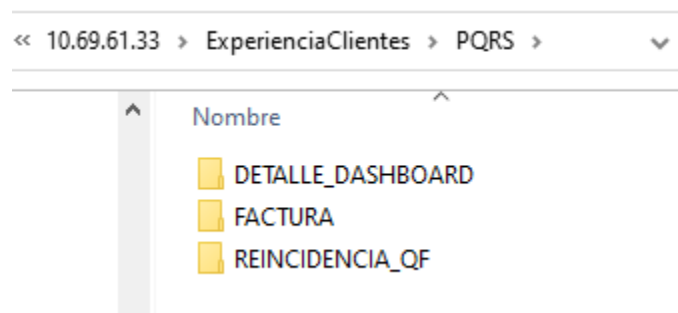
Ejecución de la tarea automática de carga del detalle de PQR's



```
C:\Windows\system32\cmd.exe
conexión exitosa
Ejecutando LIMPIEZA_DATOS_UNE.sql ...
conexión exitosa
Ejecutando LIMPIEZA_DATOS_TIGO.sql ...
conexión exitosa
Ejecutando Detalle_data_dashboard_PQRs.sql ...
conexión exitosa
Ejecutando Reincidentes_UNE.sql ...
conexión exitosa
-
```

Figura 22

Ruta y carpetas creadas para el acceso al detalle de PQR's



6 Conclusiones

La extracción de datos de diferentes fuentes y su centralización en un único servidor son de vital importancia para la toma de decisiones estratégicas y una mayor gobernanza de los datos en los diferentes procesos relacionados con PQR's de la compañía Tigo. Su implementación con herramientas como SSIS y Python permite una automatización efectiva del proceso, lo que a su vez permite una mayor eficiencia y precisión en la recopilación de datos. No obstante, la compatibilidad de los diferentes sistemas utilizados para la extracción de datos es un factor crítico en la implementación de un proceso ETL. Este tipo de problemas de compatibilidad, como en el caso de Edatel, puede requerir soluciones alternativas como el uso de lotes de registros en Python.

La limpieza y transformación de los datos es una etapa fundamental que permiten asegurar la fiabilidad y la calidad de los datos que se han recopilado a través del proceso de extracción y carga. Esta etapa es crítica en el contexto de la gestión de datos empresariales como en el caso de Tigo, ya que los datos vienen de diferentes fuentes, y es común que se puedan presentar inconsistencias al unificar la información para procesos operativos como el de liquidación de factura o procesos analíticos como la creación de KPI's y visualización de datos. Para ello, se hace necesario aplicar técnicas y herramientas que permitan homologar y normalizar los datos, a fin de asegurar su consistencia y coherencia. En este sentido, el uso de consultas SQL permite definir criterios específicos para la selección y transformación de los datos, permitiendo así obtener la información requerida de manera precisa y confiable.

La automatización de la verificación y filtrado de los registros de PQR's a través de consultas SQL permitió agilizar el proceso de liquidación de factura, lo que puede traducirse en ahorros significativos de tiempo y recursos para la empresa Tigo. Además, la creación de tablas temporales con el uso de sentencias WITH AS permitió realizar la verificación de manera más ordenada y controlada, lo que también contribuyó a la calidad de los resultados.

El KPI de reincidencia construido es un indicador clave para mejorar la experiencia del cliente respecto a las PQR's en Tigo, ya que muestra la frecuencia con la que un usuario tiene que presentar una PQR por el mismo problema. Es importante medir y monitorear este KPI para

identificar los problemas recurrentes y tomar medidas para resolverlos y mejorar la satisfacción del cliente, lo cual se hace posible a través de las herramientas de analítica desarrolladas a través de SQL y Tableau, como lo son el dashboard y la exposición del detalle de PQR's reincidentes y generadoras, para así analizar la evolución del KPI e identificar los motivos que más lo afectan.

La implementación del dashboard sobre PQR's no solo permite una mayor comprensión de los datos, sino que también facilita la toma de decisiones de negocio en la compañía Tigo. Gracias a la visualización de datos en el tablero, se pueden identificar patrones y tendencias en la información que de otra manera sería difícil de detectar, lo que permite tomar decisiones informadas y estratégicas. La preparación adecuada de los datos es crucial para la implementación de un dashboard efectivo, lo que puede implicar la combinación de diferentes fuentes de datos y la transformación de los datos para facilitar su análisis y visualización de una manera precisa y útil.

Referencias

- Perez, S. (2021). *Gestor de Base de Datos: Qué es, funcionalidades y ejemplos*, Intelequia. Disponible en: <https://intelequia.com/blog/post/2949/gestor-de-base-de-datos-qu%C3%A9-es-funcionalidades-y-ejemplos>
- IONOS Digital Guide (2022). *Oracle Database: Definición y funcionamiento*. Disponible en: <https://www.ionos.mx/digitalguide/hosting/cuestiones-tecnicas/oracle-database/>
- Wexler, S., Shaffer, J., & Cotgreave, A. (2017). *The Big Book of Dashboards: Visualizing Your Data Using Real-World Business Scenarios*. John Wiley & Sons.
- Espinosa, R. (2009). *Herramientas ETL. ¿Que son, para que valen? Productos más conocidos*. El Rincón del BI Sitio web. Disponible en: <https://churriwifi.wordpress.com/2009/12/29/11-herramientas-etl-%C2%BFque-son-para-que-valen-producto…>
- Kenny, G. (2020). What Are Your KPIs Really Measuring?. Harvard Business Review. <https://hbr.org/2020/09/what-are-your-kpis-really-measuring>