

# Prueba\_vacante\_ecosistemas

Documentación técnica – Mariana Maya Guerrero

## Contenido

Introducción.....	2
Objetivos técnicos .....	2
Tecnologías utilizadas.....	2
Requisitos de instalación.....	2
Diagrama Entidad Relación .....	3
Descripción de campos .....	3
<i>COMMERCE</i> .....	3
<i>APLICALL</i> .....	3
<i>COMISIONES</i> .....	4
<i>DESCUENTOS</i> .....	4
<i>CONSOLIDADO_SIN_COMISIONES</i> .....	4
<i>CALCULO_COMISIONES</i> .....	5
Parámetros de entrada.....	5
Ejecución .....	5
Resultados.....	5

# Introducción

La presente solución automatiza el proceso de cálculo y gestión de comisiones para BATSEJ OPEN FINANCE S.A. en el contexto de las operaciones de verificación de cuentas bancarias a través de APIs. La solución está diseñada para trabajar con una base de datos SQLite y utiliza Python para ejecutar todas las operaciones de cálculo, generación de reportes y envío de correos electrónicos.

## Objetivos técnicos

**Carga de Datos:** La solución comienza con la carga y análisis de los datos desde la base de datos SQLite ubicada en DataBase/database.sqlite. Utilizando la biblioteca sqlite3 en Python, se accede a las tablas necesarias para extraer la información relevante para el cálculo de comisiones.

**Cálculo de Comisiones:** La lógica de cálculo de comisiones se implementa según los contratos específicos de cada empresa. Se incluyen tanto comisiones fijas como variables, con un enfoque especial en el manejo de descuentos por peticiones no exitosas. Los contratos y sus respectivas tarifas se codifican en el script para garantizar flexibilidad y extensibilidad.

**Generación de Reportes:** Se crean reportes en formato Excel utilizando pandas y openpyxl, que resumen las comisiones a cobrar para cada empresa. Los reportes se almacenan en una carpeta de resultados y se generan sin índices para facilitar su uso.

**Envío de Correos Electrónicos:** La solución incluye un módulo para el envío de correos electrónicos con los reportes generados adjuntos. Se utiliza la biblioteca pywin32 (win32com.client) para enviar correos a través de Outlook de Microsoft. La información de contacto se maneja de manera dinámica para adaptarse a posibles cambios en los destinatarios.

## Tecnologías utilizadas

Python (con bibliotecas sqlite3, pandas, openpyxl y pywin32)

Microsoft Outlook (para el envío de correos electrónicos)

GitHub (para el control de versiones y entrega de la solución)

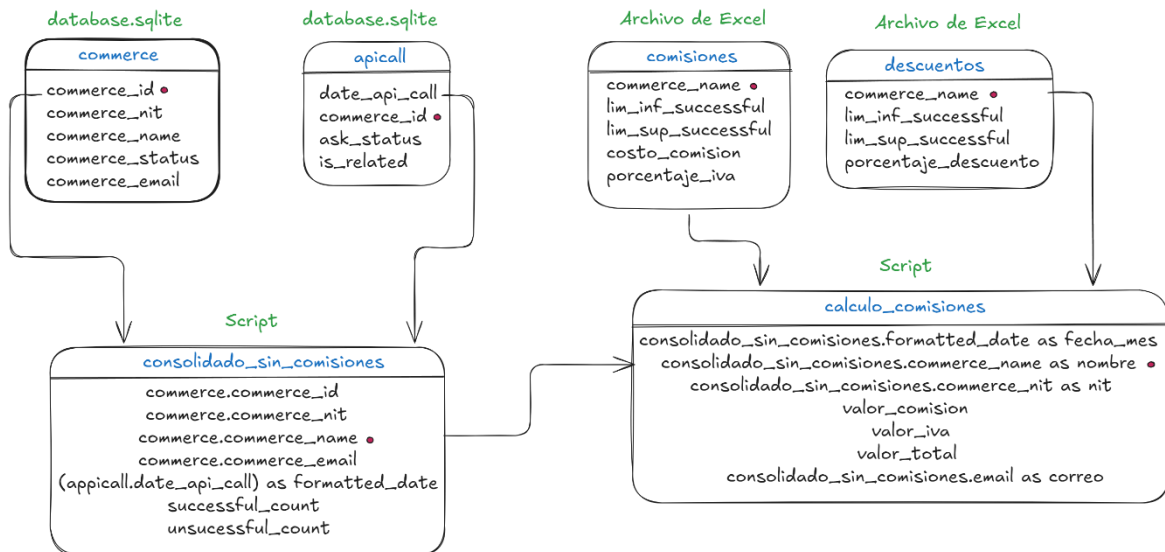
La solución está diseñada para ser ejecutable desde cualquier entorno que cumpla con los requisitos descritos, asegurando que la automatización de las comisiones sea precisa y eficiente.

## Requisitos de instalación

Para instalar los requisitos en torno a librerías de Python, es necesario ejecutar el archivo *install.bat*. Este permitirá que la solución se pueda ejecutar sin contratiempos.

# Diagrama Entidad Relación

Se construyó en Excalidraw el siguiente diagrama para entender la relación entre tablas entregadas para la solución (*database.sqlite*), generadas en base a la información suministrada (*Archivo de Excel*) y las generadas en la ejecución (*Script*).



Los puntos rojos indican las columnas de cruces.

## Descripción de campos

Los campos por tablas se identificaron y definieron de la forma:

### COMMERCE

Información de los comercios:

- **commerce\_id:** Identificación única del comercio que contrató el acceso a la API
- **commerce\_nit:** Nit del comercio que contrató el acceso a la API
- **commerce\_name:** Nombre del comercio que contrató el acceso a la API
- **commerce\_status:** Estatus *Active* o *Inactive* del contrato del comercio que contrató el acceso a la API
- **commerce\_email:** Correo electrónico del comercio que contrató el acceso a la API

### APLICALL

Registro de los llamados a la API por las empresas

- **date\_api\_call:** Fecha y hora del llamado a la API
- **commerce\_id:** Identificación única del comercio que hizo el llamado a la API
- **ask\_status:** Estado *Successful* o *Unsuccessful* al llamado de la API

- **is\_related:** Respuesta de la API al llamado (Tiene o no cuenta)

## COMISIONES

Información relacionada a las comisiones en el contrato de cada empresa

- **commerce\_name:** Nombre del comercio que contrató el acceso a la API
- **lim\_inf\_successful:** Límite inferior con el que a partir de estos llamados exitosos se cobra la comisión del campo *costo\_comision*
- **lim\_sup\_successful:** Límite superior con el que máximo a estos llamados exitosos cobra la comisión del campo *costo\_comision*
- **costo\_comision:** Comisión a cobrar si está dentro de los campos *lim\_inf\_successful* y *lim\_sup\_successful*
- **porcentaje\_iva:** Porcentaje correspondiente al impuesto IVA

## DESCUENTOS

Información relacionada a los descuentos en el contrato de cada empresa

- **commerce\_name:** Nombre del comercio que contrató el acceso a la API
- **lim\_inf\_unsuccessful:** Límite inferior con el que a partir de estos llamados no exitosos (API no da respuesta) se descuenta el porcentaje del campo *porcentaje\_descuento*
- **lim\_sup\_unsuccessful:** Límite superior con el que máximo a estos llamados no exitosos (API no da respuesta) se descuenta el porcentaje del campo *porcentaje\_descuento*
- **porcentaje\_descuento:** Porcentaje a descontar si está dentro de los campos *lim\_inf\_unsuccessful* y *lim\_sup\_unsuccessful*

## CONSOLIDADO\_SIN\_COMISIONES

Información relacionada al cálculo de operaciones exitosas y no exitosas para cada una de las llamadas de API por cada empresa contratante en estado *Active*. Este es generado por el archivo *conteo\_llamados\_api.sql*.

- **commerce\_id:** Identificación única del comercio que contrató el acceso a la API
- **commerce\_nit:** Nit del comercio que contrató el acceso a la API
- **commerce\_name:** Nombre del comercio que contrató el acceso a la API
- **commerce\_email:** Correo electrónico del comercio que contrató el acceso a la API
- **successful\_count:** Conteo de peticiones exitosas a la API por mes y empresa contratante
- **unsuccessful\_count:** Conteo de peticiones no exitosas a la API por mes y empresa contratante

## CALCULO\_COMISIONES

Información relacionada al cálculo de comisiones tras validación de reglas definidas en contratos por cada empresa. Este es generado por el archivo *calculo\_comisiones.sql*.

- **fecha\_mes:** Fecha y mes en el que se hizo el cálculo de comisión en formato yyyy-MM
- **nombre:** Nombre del comercio que contrató el acceso a la API
- **nit:** Nit del comercio que contrató el acceso a la API
- **valor\_comision:** Valor de la comisión antes de IVA y con descuento aplicado
- **valor\_iva:** Valor del impuesto IVA
- **valor\_total:** Suma del valor\_comision y el valor\_iva
- **correo:** Correo electrónico del comercio que contrató el acceso a la API

## Parámetros de entrada

En el archivo de Excel *insumos/condiciones\_por\_empresa.xlsx* existen 3 hojas de cálculo con información parametrizada para la ejecución:

1. **Comisiones:** Lo referente a la traducción del cobro de comisiones de las condiciones de contratadas por cada empresa.
2. **Descuentos:** Lo referente a la traducción del descuento por no respuesta de la API según condiciones contratadas por cada empresa.
3. **Parámetros:** Parámetros de ejecución:
  - a. **mail\_to:** Correo electrónico a quién se le enviará el archivo xlsx resultante y la información. En caso de requerir más usuario, separarlos por punto y como. Ejemplo: user1@...; user2@...
  - b. **subject:** Asunto con el que se enviará el correo.
  - c. **fecha\_inicio:** Fecha inicio de facturación en formato yyymddHHMMSS
  - d. **fecha\_fin:** Fecha fin de facturación en formato yyymddHHMMSS

## Ejecución

Tras instalar con *install.bat*, ejecutar el archivo *cobro\_comisiones.py*.

## Resultados

Envía un correo electrónico con asunto en tabla parámetros de Excel, a los destinatarios en tabla parámetros de Excel, con los valores totales a cobrar a cada empresa en estado en tabla parámetros de Excel, para el rango de fechas en la tabla parámetros de Excel.