

Informe Técnico: Sistema de Calculadora de Comisiones y PNL

Este documento proporciona un análisis detallado de la metodología, arquitectura y fórmulas utilizadas en el sistema de cálculo de PNL (Profit and Loss).

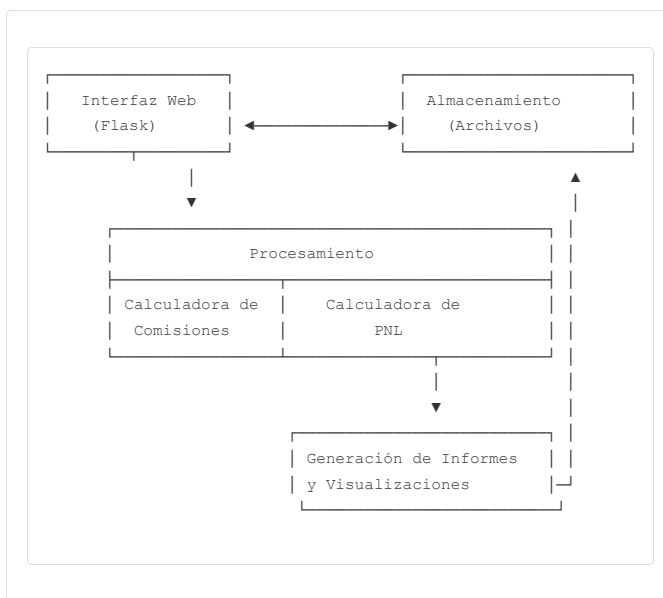
1. Arquitectura General del Sistema

El sistema está diseñado como una aplicación web Flask que permite a los usuarios cargar archivos CSV para calcular comisiones y PNL (Profit and Loss) en operaciones financieras. La arquitectura se divide en los siguientes componentes principales:

1.1 Componentes Principales

- Aplicación Web (app.py):** Interfaz de usuario con formularios para subir archivos CSV y visualizar resultados.
- Calculadora de Comisiones (comisiones_flexible.py):** Procesa archivos CSV para calcular comisiones por día.
- Calculadora de PNL (pnl_calculator.py):** Procesa archivos de compras y ventas para calcular el PNL.
- Plantillas HTML:** Interfaz gráfica para visualización de resultados.
- Generación de Gráficos:** Visualización de datos mediante matplotlib.

1.2 Diagrama de Arquitectura

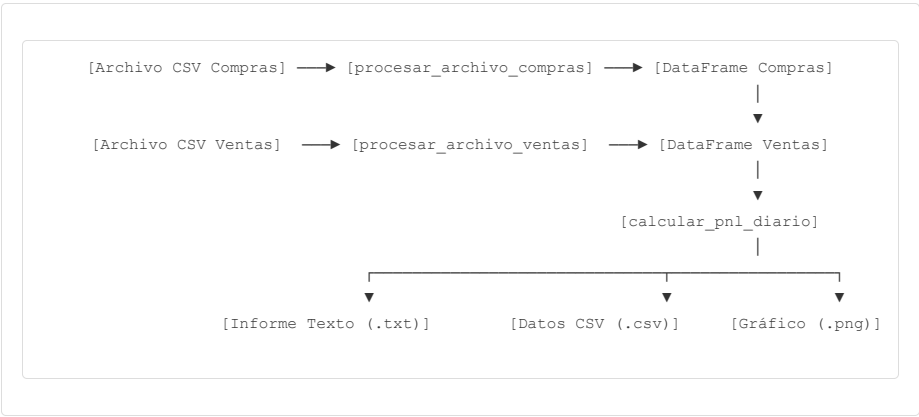


2. Flujo de Datos y Procesamiento

2.1 Flujo General

- El usuario carga archivos CSV de compras y ventas a través de la interfaz web.
- Los archivos se almacenan temporalmente con nombres únicos en el servidor.
- Los datos se procesan utilizando pandas para extraer información relevante.
- Se realizan cálculos de PNL combinando datos de compras y ventas.
- Se generan informes en formato texto, CSV y gráficos visuales.
- Los resultados se muestran en la interfaz web y están disponibles para descarga.

2.2 Diagrama de Flujo de Datos



3. Procesamiento de Datos de Compras

3.1 Función Principal

`procesar_archivo_compras(archivo_csv)`

Esta función se encarga de procesar el archivo CSV de compras y retornar un DataFrame con las compras agrupadas por día.

3.2 Pasos del Procesamiento

1. Lee el archivo CSV con precisión numérica completa (`float_precision='high'`).
2. Convierte columnas monetarias eliminando símbolos (\$ y ,) y las transforma a tipo float.
3. Calcula los USD comprados dividiendo el monto cerrado entre el precio.
4. Convierte la fecha a formato estándar y asigna el año actual si no está especificado.
5. Agrupa los datos por fecha, sumando el monto cerrado y USD comprados.
6. Calcula el precio promedio de compra para cada día.

Fórmulas Clave:

- **USD Comprado** = Monto Cerrado / Precio
- **Precio Promedio Compra** = Suma(Monto Cerrado) / Suma(USD Comprado)

```
# Código simplificado
df['USD Comprado'] = df['Monto Cerrado'] / df['Precio']

compras_por_dia = df.groupby('Fecha').agg({
    'Monto Cerrado': 'sum',
    'USD Comprado': 'sum'
}).reset_index()

compras_por_dia['Precio Promedio Compra'] = compras_por_dia['Monto Cerrado'] / compras_por_dia['USD Comprado']
```

4. Procesamiento de Datos de Ventas

4.1 Función Principal

`procesar_archivo_ventas(archivo_csv)`

Esta función procesa el archivo CSV de ventas y retorna un DataFrame con las ventas agrupadas por día.

4.2 Pasos del Procesamiento

1. Lee el archivo CSV con precisión numérica completa.
2. Filtra solo las transacciones con estado "Completed".
3. Convierte columnas monetarias a tipo float.
4. Calcula el "Valor Real" multiplicando Precio × Cantidad para cada transacción.
5. Procesa las columnas de comisiones y maneja valores vacíos.
6. Extrae la fecha del campo "Created Time".
7. Agrupa los datos por fecha, sumando valor real, cantidad y comisiones.
8. Calcula el precio promedio de venta para cada día.

Fórmulas Clave:

- **Valor Real** = Precio × Cantidad
- **Comisiones Totales** = Maker Fee + Taker Fee
- **Precio Promedio Venta** = Suma(Valor Real) / Suma(Cantidad)

```
# Código simplificado
df['Valor Real'] = df['Price'] * df['Quantity']
df['Comisiones'] = df['Maker Fee'] + df['Taker Fee']

ventas_por_dia = df.groupby('Fecha').agg({
    'Valor Real': 'sum',
    'Quantity': 'sum',
    'Comisiones': 'sum'
}).reset_index()

ventas_por_dia['Precio Promedio Venta'] = ventas_por_dia['Valor Real'] / ventas_por_dia['Quantity']
```

5. Cálculo del PNL (Profit and Loss)

5.1 Función Principal

calcular_pnl_diario(compras_df, ventas_df)

Esta función combina los datos de compras y ventas para calcular el PNL diario utilizando un método FIFO adaptado con precio promedio ponderado.

5.2 Metodología de Cálculo

El sistema utiliza un enfoque de **precio promedio ponderado** para el inventario, donde:

- Se mantiene un registro del USD remanente (stock disponible).
- Cuando hay nuevas compras, se actualiza el precio promedio ponderado del stock.
- Para las ventas, se utiliza el precio promedio actual del stock para calcular el margen.
- Las comisiones se restan directamente del PNL bruto para obtener el PNL neto.

Nota importante: El sistema implementa un método FIFO (First In, First Out) adaptado, donde los USD comprados primero son los primeros en considerarse vendidos, pero utilizando un precio promedio ponderado para el stock disponible.

5.3 Flujo del Algoritmo

1. Crear una lista de todas las fechas únicas de compras y ventas ordenadas.
2. Inicializar variables para rastrear el remanente de USD y su valor monetario.
3. Para cada día en orden cronológico:
 - Agregar compras nuevas al stock disponible.
 - Calcular el nuevo precio promedio ponderado del stock.
 - Procesar ventas del día (si existen).
 - Calcular margen y PNL basado en la diferencia entre precio de venta y precio promedio del stock.
 - Actualizar el remanente de USD y su valor monetario.
4. Acumular el PNL neto total de todos los días.

Fórmulas Clave para el Cálculo de PNL:

- **Stock Disponible** = Remanente Anterior + USD Comprado Hoy
- **Precio Promedio Stock** = (Remanente Anterior × Precio Promedio Anterior + Monto Comprado Hoy) / Stock Disponible
- **Margen** = Precio Promedio Venta - Precio Promedio Stock
- **PNL Bruto (CLP)** = Margen × USD Vendido
- **PNL Bruto (USD)** = PNL Bruto (CLP) / Precio Promedio Stock
- **PNL Neto (USD)** = PNL Bruto (USD) - Comisiones
- **Remanente USD** = Stock Disponible - USD Vendido - Comisiones

```
# Código simplificado para el cálculo de PNL diario
stock_disponible = remanente_usd + usd_comprado_hoy
if stock_disponible > 0:
    precio_promedio_stock = (remanente_monto + monto_comprado_hoy) / stock_disponible

# Cuando hay ventas
if usd_vendido_hoy > 0 and stock_disponible > 0:
    usd_vendido_efectivo = min(usd_vendido_hoy, stock_disponible)
    margen = precio_venta_hoy - precio_promedio_stock
    pnl_bruto_clp = margen * usd_vendido_efectivo
    pnl_bruto_usd = pnl_bruto_clp / precio_promedio_stock
    pnl_netto_usd = pnl_bruto_usd - comisiones_hoy
```

```
# Actualizar remanente
remanente_usd = stock_disponible - usd_vendido_efectivo - comisiones_hoy
remanente_monto = remanente_usd * precio_promedio_stock
```

6. Generación de Informes y Visualizaciones

6.1 Informe de Texto

Se genera un informe detallado en formato texto que incluye:

- Resumen diario de compras y ventas.
- Precios promedio de compra y venta por día.
- Márgenes, PNL bruto, comisiones y PNL neto por día.
- Remanente de USD después de cada día.
- PNL neto total acumulado.

6.2 Informe CSV

Se genera un archivo CSV con todos los datos calculados, incluyendo:

Columna	Descripción
Fecha	Fecha de la operación
USD Comprado	Cantidad de USD comprados en el día
Monto Compra	Monto total de la compra en moneda local
Precio Promedio Compra	Precio promedio de compra del día
USD Vendido	Cantidad de USD vendidos en el día
Monto Venta	Monto total de la venta en moneda local
Precio Promedio Venta	Precio promedio de venta del día
Precio Promedio Stock	Precio promedio ponderado del stock disponible
Margen	Diferencia entre precio de venta y precio de stock
PNL Bruto	Beneficio bruto en USD antes de comisiones
Comisiones	Total de comisiones pagadas en USD
PNL Neto	Beneficio neto en USD después de comisiones
Remanente USD	USD disponibles después de operaciones del día
Precio Promedio Remanente	Precio promedio del remanente de USD

6.3 Visualización Gráfica

Se genera un gráfico de barras que muestra:

- PNL neto diario con barras verdes (ganancias) y rojas (pérdidas).
- Valores numéricos sobre cada barra para fácil referencia.
- Título, etiquetas de ejes y formato adecuado para facilitar la interpretación.

7. Consideraciones Técnicas Importantes

7.1 Precisión Numérica

El sistema está diseñado para mantener alta precisión en cálculos financieros:

- Utilización de `float_precision='high'` en la lectura de CSV para mantener precisión completa.
- Evitar redondeos en cálculos intermedios para prevenir errores acumulativos.
- Realizar redondeo solo en la presentación final de datos (2 decimales).

7.2 Manejo de Errores

El sistema implementa manejo de errores robusto:

- Verificación de existencia de columnas necesarias en los archivos CSV.
- Manejo de valores vacíos o no numéricos en columnas de comisiones.
- Prevención de divisiones por cero en el cálculo de precios promedio.
- Validación de que no se vendan más USD de los disponibles.

7.3 Dependencias Externas

El sistema depende de las siguientes bibliotecas principales:

- **Flask:** Para la interfaz web.
- **Pandas:** Para el procesamiento y manipulación de datos.
- **Matplotlib:** Para la generación de gráficos.
- **NumPy:** Para cálculos numéricos de precisión.

8. Posibles Mejoras y Consideraciones Futuras

8.1 Mejoras Potenciales

- **Optimización de rendimiento** para archivos CSV muy grandes mediante procesamiento por lotes.
- **Interfaz de usuario mejorada** con filtros y opciones de visualización adicionales.
- **Soporte para diferentes métodos de contabilidad de inventario** (LIFO, Promedio Ponderado, etc.).
- **Exportación a formatos adicionales** como Excel o PDF con formato mejorado.
- **Almacenamiento de históricos** para comparar resultados entre diferentes periodos.

8.2 Consideraciones para Implementaciones Específicas

Al adaptar este sistema a casos de uso específicos, considere:

- Ajustes en el formato de los archivos CSV de entrada para acomodar diferentes fuentes de datos.
- Modificaciones en el cálculo del PNL para cumplir con requisitos contables específicos.
- Adaptaciones para manejar diferentes monedas o pares de trading.
- Integración con sistemas contables o plataformas de trading existentes.

Este informe ha sido generado automáticamente basado en el análisis del código fuente del sistema de calculadora de PNL.

Fecha de generación: 11/4/2025, 2:21:16