

Servidor MCP de Análisis Financiero

Un servidor completo de Model Context Protocol (MCP) que proporciona herramientas avanzadas de análisis técnico y fundamental para mercados financieros. Este servidor se integra con Claude Desktop, smolagents y otros clientes MCP para ofrecer análisis sofisticado de estrategias de trading, backtesting de rendimiento y capacidades de escaneo de mercado.

Tabla de Contenidos

- [Descripción General](#)
- [Arquitectura](#)
- [Instalación](#)
- [Configuración](#)
- [Herramientas Disponibles](#)
 - [Herramientas de Análisis de Estrategias](#)
 - [Herramientas de Backtesting de Rendimiento](#)
 - [Herramientas de Escaneo de Mercado](#)
 - [Herramientas de Análisis Fundamental](#)
- [Referencia de Herramientas](#)
- [Funciones de Utilidad](#)
- [Ejemplos de Uso](#)
- [Guía de Integración](#)

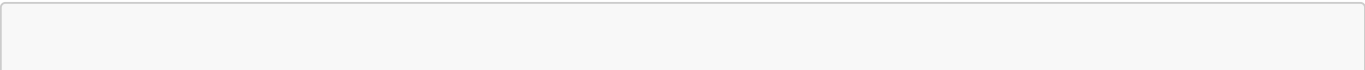
Descripción General

El Servidor Financiero MCP implementa **cinco estrategias distintas de análisis técnico** más **análisis fundamental** con capacidades completas de backtesting de rendimiento. Construido sobre el framework FastMCP, proporciona una interfaz estandarizada para que asistentes de IA y herramientas de automatización accedan a análisis financiero sofisticado.

Características Clave

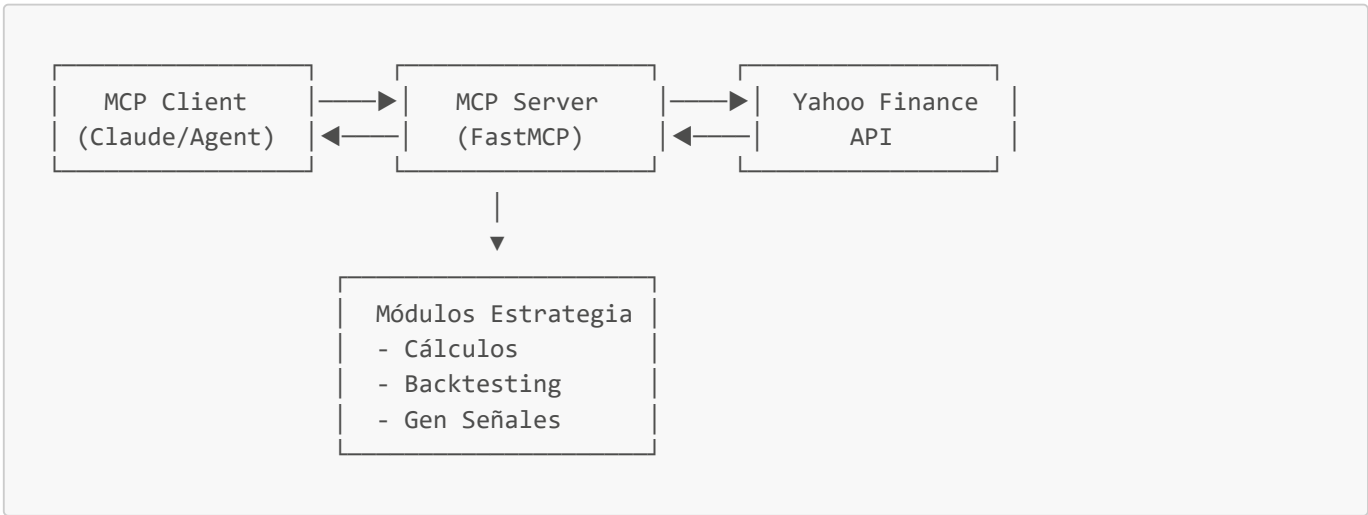
Característica	Descripción
5 Estrategias Técnicas	Bollinger-Fibonacci, MACD-Donchian, Connors RSI, Dual MA, Bollinger Z-Score
Análisis Fundamental	Análisis de estado de resultados, balance general y flujo de caja
Backtesting de Rendimiento	Compara retornos de estrategia vs buy-and-hold con métricas detalladas
Escáner de Mercado	Analiza múltiples acciones simultáneamente con clasificaciones
Evaluación de Riesgo	Cálculos de volatilidad, ratios de Sharpe, caída máxima
Generación de Señales	Recomendaciones COMPRA/MANTENER/VENTA en tiempo real con puntajes de confianza

Arquitectura



```
server/
├── main.py                # Punto de entrada del servidor MCP
├── strategies/            # Módulos de estrategias de trading
│   ├── __init__.py
│   ├── bollinger_fibonacci.py    # Bandas de Bollinger + Retroceso de Fibonacci
│   ├── bollinger_zscore.py       # Bandas de Bollinger + Reversión a la Media Z-Score
│   ├── macd_donchian.py          # MACD + Breakout Canal Donchian
│   ├── connors_zscore.py         # Connors RSI + Z-Score Combinado
│   ├── dual_moving_average.py    # Estrategia de Cruce EMA/SMA
│   ├── fundamental_analysis.py   # Análisis de Estados Financieros
│   ├── performance_tools.py      # Backtesting y Comparación de Rendimiento
│   ├── comprehensive_analysis.py # Informes Multi-Estrategia
│   └── unified_market_scanner.py # Escáner de Mercado Multi-Acción
└── utils/
    ├── __init__.py
    └── yahoo_finance_tools.py    # Obtención de datos y cálculos de indicadores
```

Flujo de Datos



Instalación

Prerrequisitos

- Python 3.10+
- Conexión a internet (acceso a datos de Yahoo Finance)

Dependencias

```
pip install mcp fastmcp yfinance pandas numpy scipy
```

Inicio Rápido

```
# Ejecutar el servidor directamente
python server/main.py
```

```
# O con el gestor de paquetes UV
uv run python server/main.py
```

Configuración

Integración con Claude Desktop

Agregar a `claude_desktop_config.json`:

Windows: `%APPDATA%\Claude\claude_desktop_config.json`
macOS: `~/Library/Application Support/Claude/claude_desktop_config.json`

```
{
  "mcpServers": {
    "finance-tools": {
      "command": "python",
      "args": ["/ruta/a/server/main.py"]
    }
  }
}
```

Extensión MCP de VS Code

Agregar a `.vscode/mcp.json`:

```
{
  "servers": {
    "finance-tools": {
      "command": "python",
      "args": ["D:\\ruta\\a\\server\\main.py"]
    }
  }
}
```

Herramientas Disponibles

Herramientas de Análisis de Estrategias

Estas herramientas calculan puntajes y señales en tiempo real basados en datos de mercado actuales.

Herramienta	Estrategia	Rango de Señal	Caso de Uso
<code>calculate_bollinger_fibonacci_score</code>	Bollinger + Fibonacci	-100 a +100	Soporte/Resistencia
<code>calculate_bollinger_z_score</code>	Bollinger + Z-Score	-100 a +100	Reversión a la Media

Herramienta	Estrategia	Rango de Señal	Caso de Uso
<code>calculate_combined_score_macd_donchian</code>	MACD + Donchian	-100 a +100	Momentum/Breakout
<code>calculate_connors_rsi_score_tool</code>	Connors RSI	0 a 100	Momentum Corto Plazo
<code>calculate_combined_connors_zscore_tool</code>	Connors + Z-Score	-100 a +100	Momentum Combinado
<code>analyze_dual_ma_strategy</code>	Dual Moving Average	COMPRA/MANTENER/VENTA	Seguimiento de Tendencia

Herramientas de Backtesting de Rendimiento

Estas herramientas ejecutan backtests históricos y comparan el rendimiento de la estrategia vs buy-and-hold.

Herramienta	Descripción	Métricas Clave
<code>analyze_bollinger_fibonacci_performance</code>	Backtest estrategia BB-Fib	Retorno %, Sharpe, DD Máx
<code>analyze_bollinger_zscore_performance</code>	Backtest estrategia BB-ZScore	Retorno %, Tasa de Éxito
<code>analyze_macd_donchian_performance</code>	Backtest MACD-Donchian	Retorno Exceso, Trades
<code>analyze_connors_zscore_performance</code>	Backtest Connors-ZScore	Estrategia vs B&H
<code>analyze_dual_ma_strategy</code>	Backtest Dual MA	Estadísticas Golden/Death Cross

Herramientas de Escaneo de Mercado

Herramienta	Descripción	Formatos de Salida
<code>market_scanner</code>	Escáner multi-acción unificado	detailed, summary, executive
<code>generate_comprehensive_analysis_report</code>	Acción única, todas las estrategias	Informe markdown completo

Herramientas de Análisis Fundamental

Herramienta	Descripción	Fuente de Datos
<code>fundamental_analysis</code>	Análisis financiero completo	yfinance financials
<code>get_financial_statement_index</code>	Listar métricas disponibles	Income, Balance, Cash Flow

Referencia de Herramientas

1. Bandas de Bollinger + Retroceso de Fibonacci

Herramienta: `calculate_bollinger_fibonacci_score`

Lógica de la Estrategia:

- Combina Bandas de Bollinger (volatilidad) con niveles de retroceso de Fibonacci (soporte/resistencia)
- Identifica puntos potenciales de reversión donde el precio encuentra tanto niveles BB como Fib

Componentes del Puntaje (ponderados):

Componente	Peso	Descripción
Posición Banda Bollinger	30%	Indicador %B (rango 0-1)
Evaluación Volatilidad	15%	Ancho BB y expansión
Interacción Fibonacci	35%	Proximidad a niveles Fib clave
Momentum de Precio	20%	Indicador tipo RSI de momentum

Zonas de Señal:

+60 a +100: Compra Fuerte

+20 a +60: Compra Moderada

-20 a +20: Mantener

-60 a -20: Venta Moderada

-100 a -60: Venta Fuerte

Parámetros:

```
calculate_bollinger_fibonacci_score(  
    ticker: str,          # Símbolo de acción (ej., "AAPL")  
    period: str = "1y",   # Período de datos: 1d,5d,1mo,3mo,6mo,1y,2y,5y,10y,ytd,max  
    window: int = 20,     # Período de Banda Bollinger  
    num_std: int = 2,     # Desviaciones estándar para bandas  
    window_swing_points: int = 10, # Ventana de detección de punto swing  
    fibonacci_levels: List = [0, 0.236, 0.382, 0.5, 0.618, 0.786, 1]  
)
```

2. MACD + Canal Donchian

Herramienta: `calculate_combined_score_macd_donchian`

Lógica de la Estrategia:

- MACD identifica momentum y dirección de tendencia
- Canales Donchian identifican breakouts y límites de rango
- Combinados para señales de breakout confirmadas por momentum

Componentes del Puntaje:

Componente	Peso	Descripción
Línea MACD vs Señal	40%	Momentum de cruce
MACD vs Línea Cero	30%	Dirección de tendencia

Componente	Peso	Descripción
Histograma MACD	30%	Aceleración de momentum
Posición Donchian	50%	Detección de breakout de canal

Parámetros:

```
calculate_combined_score_macd_donchian(  
    symbol: str,  
    period: str = "1y",  
    fast_period: int = 12,      # EMA rápido MACD  
    slow_period: int = 26,     # EMA lento MACD  
    signal_period: int = 9,    # Línea de señal MACD  
    window: int = 20           # Período de canal Donchian  
)
```

3. Connors RSI + Z-Score

Herramienta: `calculate_combined_connors_zscore_tool`

Lógica de la Estrategia:

- Connors RSI: Indicador de reversión a la media de corto plazo
- Z-Score: Desviación estadística de la media
- Combinados para señales de reversión de alta probabilidad

Componentes de Connors RSI:

Componente	Peso	Descripción
RSI de Precio	33.3%	RSI tradicional de cierres
RSI de Racha	33.3%	RSI de rachas arriba/abajo
Percentil Rango	33.3%	Percentil de tasa de cambio

Pesos del Puntaje Combinado:

- Connors RSI: 70%
- Z-Score: 30%

Interpretación de Señales:

```
CRSI < 20: Sobreventa (Potencial Compra)  
CRSI > 80: Sobrecompra (Potencial Venta)  
Z-Score < -2: Extremadamente Sobrevendido  
Z-Score > +2: Extremadamente Sobrecomprado
```

Parámetros:

```

calculate_combined_connors_zscore_tool(
    symbol: str,
    period: str = "1y",
    rsi_period: int = 3,      # Período Connors RSI
    streak_period: int = 2,  # Período RSI de racha
    rank_period: int = 100,  # Lookback de percentil rango
    zscore_window: int = 20, # Ventana de cálculo Z-Score
    connors_weight: float = 0.7,
    zscore_weight: float = 0.3
)

```

4. Cruce de Media Móvil Dual

Herramienta: `analyze_dual_ma_strategy`

Lógica de la Estrategia:

- Estrategia clásica de seguimiento de tendencia
- Golden Cross (50 > 200): Señal alcista
- Death Cross (50 < 200): Señal bajista

Generación de Señales:

COMPRA: MA Corto cruza POR ENCIMA de MA Largo
 VENTA: MA Corto cruza POR DEBAJO de MA Largo
 MANTENER: Sin cruce reciente

Parámetros:

```

analyze_dual_ma_strategy(
    symbol: str,
    period: str = "2y",
    short_period: int = 50,  # Período MA corto (días)
    long_period: int = 200,  # Período MA largo (días)
    ma_type: str = "EMA"    # "SMA" o "EMA"
)

```

Métricas de Salida:

- Retorno Total vs Buy & Hold
- Retorno Exceso
- Tasa de Éxito
- Total de Trades (cruces)
- Ratio de Sharpe
- Caída Máxima

5. Bollinger Z-Score (Reversión a la Media)

Herramienta: `calculate_bollinger_z_score`

Lógica de la Estrategia:

- Reversión a la media estadística pura
- Z-Score mide desviaciones estándar de la media móvil
- Comprar cuando sobrevendido (Z bajo), vender cuando sobrecomprado (Z alto)

Zonas de Señal:

```
Z < -2.0: Compra Fuerte (sobrevendido)
Z < -1.0: Compra
-1 < Z < 1: Mantener
Z > 1.0: Venta
Z > 2.0: Venta Fuerte (sobrecomprado)
```

Parámetros:

```
calculate_bollinger_z_score(
    symbol: str,
    period: str = "1y",
    window: int = 20 # Ventana de cálculo Z-Score
)
```

6. Escáner de Mercado

Herramienta: `market_scanner`

Descripción: Analiza múltiples acciones simultáneamente usando las 5 estrategias, las clasifica por rendimiento y fuerza de señal, y proporciona recomendaciones estructuradas.

Parámetros:

```
market_scanner(
    symbols: str, # Separado por comas: "AAPL,MSFT,GOOGL"
    period: str = "1y",
    output_format: str = "detailed" # "detailed", "summary", "executive"
)
```

La Salida Incluye:

- Resumen Ejecutivo con perspectivas de mercado
- Análisis individual de acciones con todas las estrategias
- Comparación de rendimiento vs buy-and-hold
- Consenso de señales y fuerza
- Evaluación de riesgo
- Recomendaciones clasificadas

7. Análisis Fundamental

Herramienta: `fundamental_analysis`

Descripción: Analiza estados financieros de la empresa incluyendo estado de resultados, balance general y estado de flujo de caja.

Parámetros:

```
fundamental_analysis(  
    symbol: str,  
    period: str = "3y"      # Años de datos financieros  
)
```

Métricas Analizadas:

Categoría	Métricas
Rentabilidad	Ingresos, Ingreso Neto, Márgenes, ROE, ROA
Crecimiento	Crecimiento de Ingresos, Crecimiento de Ganancias
Liquidez	Ratio Corriente, Ratio Rápido
Apalancamiento	Deuda-Capital, Cobertura de Intereses
Flujo de Caja	CF Operativo, Flujo de Caja Libre, CapEx
Valuación	P/E, P/B, P/S (cuando disponible)

Funciones de Utilidad

Ubicadas en `utils/yahoo_finance_tools.py`:

Obtención de Datos

```
fetch_data(ticker: str, period: str) -> pd.DataFrame
```

Indicadores Técnicos

```
# Bandas de Bollinger  
calculate_bollinger_bands(data, ticker, period, window, num_std) -> pd.DataFrame  
  
# Niveles de Fibonacci  
find_swing_points(data, window) -> pd.DataFrame  
calculate_fibonacci_levels(swing_high, swing_low, levels) -> Dict  
  
# Cálculos RSI  
rsi(series, period) -> pd.Series  
streak_rsi(series, period) -> pd.Series  
percent_rank(series, period) -> pd.Series  
  
# Connors RSI  
calculate_connors_rsi_score(symbol, period, rsi_period, streak_period, rank_period) ->
```

Tuple

```
# Z-Score
```

```
calculate_zscore_indicator(symbol, period, window) -> Tuple
```

```
# MACD
```

```
calculate_macd_score(symbol, period, fast, slow, signal) -> float
```

```
# Canales Donchian
```

```
calculate_donchian_channel_score(symbol, period, window) -> float
```

Ejemplos de Uso

Prompts de Claude Desktop

Análisis de Acción Única:

Analiza TSLA usando la estrategia Bollinger-Fibonacci con un período de 1 año

Comparación Multi-Acción:

Usa market scanner con símbolos "AAPL, MSFT, GOOGL, META, NVDA" con period "1y" y output_format "detailed"

Análisis de Sector:

Escanea estas acciones bancarias: JPM, BAC, WFC, C, GS, MS, USB, PNC, TFC, COF

Análisis Completo:

Para AAPL:

- Ejecuta analyze_bollinger_fibonacci_performance con período de 1 año
- Ejecuta analyze_macd_donchian_performance con período de 1 año
- Ejecuta analyze_connors_zscore_performance con parámetros predeterminados
- Ejecuta analyze_dual_ma_strategy con 50/200 EMA
- Compila resultados en un informe completo

Uso Directo en Python

```
from mcp import ClientSession, StdioServerParameters
from mcp.client.stdio import stdio_client

async def analyze_stock():
    server_params = StdioServerParameters(
        command="python",
        args=["server/main.py"]
```

```
)

async with stdio_client(server_params) as (read, write):
    async with ClientSession(read, write) as session:
        await session.initialize()

        # Llamar una herramienta
        result = await session.call_tool(
            "analyze_bollinger_fibonacci_performance",
            {"symbol": "AAPL", "period": "1y", "window": 20}
        )
        print(result.content[0].text)
```

Guía de Integración

Con smolagents (stock_analyzer_bot)

El servidor está diseñado para funcionar con el `stock_analyzer_bot` que usa smolagents:

```
from stock_analyzer_bot.tools import (
    bollinger_fibonacci_analysis,
    macd_donchian_analysis,
    connors_zscore_analysis,
    dual_moving_average_analysis,
)

# Estas herramientas llaman internamente al servidor MCP
result = bollinger_fibonacci_analysis("AAPL", "1y")
```

Con Backend FastAPI

El `stock_analyzer_bot/api.py` expone estas herramientas vía REST:

```
# Iniciar el servidor API
uvicorn stock_analyzer_bot.api:app --reload --port 8000

# Endpoints disponibles:
# POST /technical - Análisis técnico de acción única
# POST /scanner - Escáner de mercado multi-acción
# POST /fundamental - Análisis fundamental
# POST /multisector - Análisis entre sectores
# POST /combined - Técnico + Fundamental combinado
```

Con Frontend Streamlit

```
streamlit run streamlit_app.py
```

Proporciona una interfaz web para todos los tipos de análisis.

Glosario de Métricas de Rendimiento

Métrica	Descripción	Valor Bueno
Retorno Total	Retorno acumulativo de estrategia	> Buy & Hold
Retorno Exceso	Retorno por encima de buy-and-hold	> 0%
Ratio de Sharpe	Retorno ajustado por riesgo	> 1.0
Caída Máxima	Mayor declive pico a valle	> -20%
Tasa de Éxito	Porcentaje de trades rentables	> 50%
Volatilidad	Desviación estándar anualizada	Menor = menos riesgo

Manejo de Errores

El servidor maneja errores comunes con gracia:

- **Símbolo Inválido:** Retorna mensaje de error con sugerencia
- **Sin Datos:** Maneja datos faltantes de Yahoo Finance
- **Errores de Cálculo:** Retorna resultados parciales cuando es posible
- **Problemas de Red:** Manejo de timeout para llamadas API

Historial de Versiones

Versión	Cambios
1.0.0	Lanzamiento inicial con 5 estrategias técnicas
1.1.0	Agregado escáner de mercado unificado
1.2.0	Agregadas herramientas de análisis fundamental
1.3.0	Optimizaciones de rendimiento, mejor manejo de errores

Licencia

Este servidor se proporciona para fines educativos e investigación. Siempre verifica los resultados de análisis y consulta profesionales financieros antes de tomar decisiones de inversión.

Construido con [FastMCP](#) y [yfinance](#)