¡Excelente proyecto! Construir tu propia plataforma de trading es una meta ambiciosa y muy formativa. Te permitirá entender a fondo cómo funcionan los mercados y las estrategias. Aquí tienes una guía detallada y realista para que puedas empezar.

La plataforma se puede dividir en varios componentes clave. Piénsalo como construir un coche: necesitas un motor, un chasis, un sistema de combustible y un volante.

**Componentes Clave de tu Plataforma ⚙️**

Una plataforma de trading algorítmico robusta consta de varias capas que trabajan juntas:

1. **Adquisición de Datos:** El combustible. Es el servicio que te proporciona los precios y volúmenes del mercado.
2. **Procesamiento y Análisis:** El motor. Aquí es donde tu código recibe los datos crudos y calcula los indicadores (MACD, RSI, etc.).
3. **Lógica de Estrategia y Señales:** El cerebro. Define las reglas que, basadas en los indicadores, deciden cuándo comprar o vender.
4. **Ejecución de Órdenes:** La conexión con el mundo real. Envía tus órdenes de compra/venta a un bróker.
5. **Interfaz y Monitorización:** El salpicadero. Te permite ver qué está pasando, el estado de tus operaciones y el rendimiento de tu sistema.

**Roadmap para Construir tu Plataforma**

Aquí tienes un plan de acción paso a paso, con las tecnologías y herramientas más comunes en la industria.

**Paso 1: Adquisición de Datos (El Combustible)**

Necesitas una **API (Interfaz de Programación de Aplicaciones)** que te envíe los datos del mercado. Tienes varias opciones:

* **Tiempo Real vs. Retardo:** Las APIs en tiempo real son más caras. Para empezar, una con 15 minutos de retardo o datos al final del día (*end-of-day*) puede ser suficiente y, a menudo, gratuita.
* **Proveedores de API de Datos:**
  + **Alpha Vantage:** Muy popular para empezar. Tiene un plan gratuito generoso que es ideal para desarrollo y pruebas con datos históricos y con algo de retardo.
  + **Polygon.io:** Ofrece datos de muy alta calidad y baja latencia. Es de pago, pero es un estándar en la industria para cuando quieras pasar a un nivel más profesional.
  + **Finnhub:** Otra excelente opción con planes gratuitos y de pago, que cubre acciones, cripto y forex.
  + **EOD Historical Data (EODHD):** Muy buena para datos históricos y fundamentales.

**Tu objetivo aquí:** Escribir un script (preferiblemente en Python) que se conecte a una de estas APIs y descargue los datos de precios (apertura, máximo, mínimo, cierre y volumen) de una acción para un período determinado.

**Paso 2: Procesamiento y Análisis Técnico (El Motor)**

Una vez que tienes los datos, necesitas calcular tus indicadores. **Python** es el lenguaje rey para esto por su ecosistema de librerías.

* **Librerías Esenciales:**
  + **Pandas:** Es la herramienta fundamental para manejar los datos en series de tiempo y tablas (DataFrames). Indispensable.
  + **NumPy:** Para cálculos numéricos rápidos y eficientes.
  + **TA-Lib** o **Pandas TA:** Estas son las librerías mágicas. Contienen funciones pre-construidas para calcular decenas de indicadores técnicos como:
    - pandas\_ta.macd()
    - pandas\_ta.rsi()
    - pandas\_ta.sma() (Media Móvil Simple)
    - pandas\_ta.ema() (Media Móvil Exponencial)

**Tu objetivo aquí:** Cargar los datos que descargaste en el paso 1 en un DataFrame de Pandas y usar Pandas TA para añadir columnas nuevas con los valores calculados de RSI, MACD y un par de medias móviles.

**Paso 3: Lógica de Estrategia y Generación de Señales (El Cerebro)**

Aquí es donde defines tus reglas de trading. Esto es único para cada trader. Un ejemplo muy simple podría ser:

* **Señal de Compra:**
  1. El RSI cruza por encima de 30 (saliendo de la sobreventa).
  2. Y la línea MACD cruza por encima de su línea de señal.
* **Señal de Venta:**
  1. El RSI cruza por debajo de 70 (saliendo de la sobrecompra).
  2. Y la línea MACD cruza por debajo de su línea de señal.

**Tu objetivo aquí:** Escribir una función en Python que recorra tus datos (con los indicadores ya calculados) y genere una señal ("COMPRAR", "VENDER", "MANTENER") para cada vela o período de tiempo.

**Paso 4: Backtesting y Ejecución de Órdenes**

Antes de arriesgar dinero real, debes probar tu estrategia con datos históricos. Esto se llama **backtesting**.

* **Backtesting:** Simula cómo le habría ido a tu estrategia en el pasado. Te ayuda a entender su rentabilidad potencial, su riesgo y sus puntos débiles. Librerías como **backtesting.py** o **Zipline** pueden ayudarte a hacerlo de forma estructurada.
* **Conexión con un Bróker (Paper/Live Trading):** Para hacer trading de verdad, necesitas una cuenta con un bróker que ofrezca una API.
  + **Alpaca:** Muy popular para trading algorítmico. No tiene comisiones de compra-venta de acciones y su API es muy amigable para desarrolladores. Ofrece un entorno de "paper trading" (simulado con dinero ficticio) que es perfecto para probar tu plataforma sin riesgo.
  + **Interactive Brokers (IBKR):** El estándar de oro para traders avanzados y profesionales. Su API es extremadamente potente y da acceso a mercados de todo el mundo, aunque es más compleja de usar.
  + **IG:** También ofrece API para automatizar operaciones.

**Tu objetivo aquí:** Primero, implementar un backtest simple para tu estrategia. Segundo, crear una cuenta de "paper trading" en Alpaca y escribir el código que envíe una orden de compra o venta a través de su API cuando tu lógica genere una señal.

**Consideraciones Muy Importantes ⚠️**

* **Costes:** Los datos en tiempo real de alta calidad y la ejecución de muchas operaciones pueden tener costes asociados (suscripciones a APIs, comisiones del bróker).
* **Complejidad Técnica:** Empezar es relativamente sencillo, pero crear un sistema robusto que funcione 24/7 sin fallos es un desafío de ingeniería de software significativo.
* **Riesgo Financiero:** El trading algorítmico no es una garantía de beneficios. Las estrategias que funcionan bien en el backtesting pueden fallar en el mercado real. **Nunca operes con dinero que no te puedas permitir perder.** Empieza siempre con paper trading.
* **Latencia:** Para estrategias de alta frecuencia, el retardo (latencia) de tus datos y de la ejecución de tus órdenes es crítico. Para estrategias a más largo plazo (horas, días), no es tan crucial.

Construir esta plataforma es un viaje increíble. Empleza poco a poco, valida cada paso y aprende de tus errores. ¡Mucha suerte con tu proyecto!