

■ TRADING PHANTOM

Enterprise-Grade Algorithmic Trading Platform with ML Intelligence

Versión: 1.1.0

Fecha: January 07, 2026

Licencia: MIT

Lenguaje: Python 3.10+

Plataforma: MetaTrader 5 | Windows

Un bot de trading completamente automatizado con inteligencia artificial, backtesting profesional y Knowledge Base para futuras IAs.

■ Tabla de Contenidos

1. Introducción
2. Características principales
3. Requisitos técnicos
4. Instalación
5. Estructura del proyecto
6. Arquitectura y diseño
7. Sistema de Machine Learning
8. Knowledge Base
9. API REST
10. Backtesting
11. Empaquetado
12. Solución de problemas
13. Roadmap

1. Introducción

Trading Phantom es una plataforma modular de trading algorítmico construida en Python, diseñada para operar en MetaTrader 5 con inteligencia artificial integrada. El proyecto combina automatización completa, machine learning avanzado, backtesting profesional y una interfaz web moderna en una solución empresarial.

Casos de uso:

- **Traders profesionales:** Automatiza estrategias, backtesta y monitorea 24/7
- **Analistas cuantitativos:** Experimenta con indicadores y ML sin código repetitivo
- **Investigadores de IA:** Infraestructura lista para integrar LSTM, RL, Transformers
- **Desarrolladores:** API REST + modularidad para crear bots personalizados
- **Educación:** Aprende trading algorítmico con código profesional y documentado

2. Características Principales

■ Bot de Trading Inteligente

Loop automático configurable con indicadores técnicos (EMA, MACD, RSI), validación de riesgo, conexión MT5 y logging profesional.

■ Machine Learning Integrado

RandomForest entrenado con datos históricos, 7 indicadores como features, Knowledge Base que captura 8 tipos de aprendizaje.

■ UI Profesional

Dashboard con KPIs en tiempo real, panel ML, logbox elegante, botón de shutdown con confirmación, diseño responsivo.

■ Backtesting Visual

Simulación numérica, gráficos interactivos, métricas detalladas (Sharpe, Drawdown, Win Rate), exportación de resultados.

■ API REST Completa

20+ endpoints para control del bot, backtesting, ML, exportación de datos y acceso a Knowledge Base.

■ Docker & Escalabilidad

docker-compose.yml con Postgres + Flask, base de datos persistente, healthchecks automatizados.

■ Empaquetado Profesional

EXE Windows con PyInstaller, instalador Windows con Inno Setup, self-contained sin dependencias externas.

3. Requisitos Técnicos

Requisito	Versión	Descripción
Windows	10 o superior	Sistema operativo soportado
Python	3.10+	Lenguaje de programación
Git	Opcional	Para clonar repositorio
MetaTrader 5	Opcional	Solo para operar en vivo
Inno Setup	Opcional	Solo para crear instalador

4. Instalación

Opción 1: Automática (Recomendado)

1. Abre **INSTALL.bat** (doble-click)
 - Crea venv automáticamente
 - Instala dependencias
2. Abre **RUN.bat** (doble-click)
 - Inicia servidor Flask
 - Abre UI en <http://127.0.0.1:5000>

Opción 2: Manual (Desarrollo)

```
python -m venv .venv
.\.venv\Scripts\Activate.ps1
pip install -r requirements.txt
pip install -r requirements-dev.txt
python -m trading_phantom.main --debug
```

5. Estructura del Proyecto

El proyecto sigue el patrón **src-layout**, organizando el código principal en `src/trading_phantom/` con responsabilidades bien definidas:

src/trading_phantom/core/: Orquestación principal (Orchestrator)

src/trading_phantom/modules/: Strategy, RiskManager, Trader, DataLoader

src/trading_phantom/analytics/: ML Pipeline, Knowledge Base, Transfer Learning

src/trading_phantom/api/: REST API Blueprints (bot, backtest, analytics)

src/trading_phantom/mt5/: Integración MetaTrader 5

src/trading_phantom/backtest/: Engine de backtesting

tests/: Tests unitarios e integración

docs/: Documentación profesional

scripts/: Scripts de desarrollo y empaquetado

6. Arquitectura y Diseño

Trading Phantom utiliza una arquitectura de **capas bien definidas** que facilita testing, escalabilidad y mantenimiento:

Presentation Layer: HTML/CSS/JS (Flask templates) - Dashboard y UI

API Layer: Flask Blueprints - REST endpoints

Business Logic Layer: Orchestrator, Strategy, ML Pipeline

Data Access Layer: MT5Connector, Database, Cache

Patrones de Diseño Utilizados:

- Orchestrator Pattern - Coordina el flujo principal
- Strategy Pattern - Estrategias intercambiables
- Dependency Injection - Componentes desacoplados
- Adapter Pattern - Adaptación a diferentes interfaces
- Repository Pattern - Abstracción de acceso a datos

7. Sistema de Machine Learning

Trading Phantom integra un sistema de **ML modular y escalable** que aprende de los datos históricos de trading sin afectar la estrategia base.

Componentes ML:

RandomForest Classifier: Modelo principal de predicción

7 Features: EMA, MACD, RSI, cambio precio, volumen, volatilidad, tendencia

Database: SQLite (local) o Postgres (producción)

Collector: Ingesta automática de trades y backtests

Flujo de ML:

1. **Recolecta de datos:** Bot ejecuta trades → auto-ingesta en DB
2. **Entrenar modelo:** RandomForest.fit() con ≥ 30 trades
3. **Activar ML:** config.yaml ml.enabled=true
4. **Predicción:** Cada vela, ML valida señal con probabilidad

8. Knowledge Base

La **Knowledge Base** es un sistema único que captura el aprendizaje completo del RandomForest y lo expone de forma modular para que futuras IAs (LSTM, RL, Transformers) puedan consumir sin reentrenar.

8 Tipos de Conocimiento Almacenados:

#	Tipo	Archivo	Contenido
1	Feature Importance	feature_importance.json	Top 5 features ranking
2	Feature Embeddings	feature_embeddings.json	Mean, std, min, max
3	Correlation Matrix	correlation_matrix.json	Feature relationships
4	Decision Patterns	decision_patterns.json	Reglas del árbol
5	Performance Metrics	performance_metrics.json	Accuracy, precision, recall
6	Training Data Stats	feature_stats.json	Distribución dataset
7	Trade Patterns	winners_losers.json	Análisis trades
8	Model Serialization	random_forest.pkl	Modelo guardado

Ubicación: src/trading_phantom/data/knowledge_base/

9. API REST

Trading Phantom expone **20+ endpoints REST** para integración, automatización y acceso programático.

Categorías de Endpoints:

- **Bot:** /api/bot/start, /api/bot/stop, /api/bot/status
- **Backtest:** /api/backtest (POST/GET)
- **ML:** /api/analytics/ml/train, /api/analytics/ml/predict
- **Exportación:** /api/analytics/export/trades, /api/analytics/export/backtests
- **Knowledge Base:** /api/knowledge/* (8 endpoints)
- **Logs:** /api/logs

10. Backtesting Visual

El módulo de backtesting permite validar estrategias con datos históricos reales antes de operar en vivo.

Características:

- ✓ Simulación numérica completa
- ✓ Gráficos interactivos (Equity curve, Drawdown)
- ✓ Métricas profesionales (Sharpe, Sortino, Calmar, Max DD, Win Rate)
- ✓ Exportación en JSON/CSV/Parquet
- ✓ Comparación entre estrategias (A/B testing)
- ✓ Análisis individual de trades

11. Empaquetado y Distribución

Generar ejecutable .exe:

```
.\scripts\build_exe.ps1 # Sin consola (usuario final)  
.\scripts\build_exe.ps1 -console # Con consola (debugging)
```

Crear instalador Windows:

```
.\scripts\build_installer.ps1
```

→ Resultado: Setup-TradingPhantom-v1.1.0.exe

Beneficios: Self-contained (sin dependencias externas), autostart opcional, instalación limpia, fácil desinstalación.

12. Roadmap

■ Completado (v1.1.0)

- ✓ Bot de trading con indicadores (EMA, MACD, RSI)
- ✓ Backtesting visual con métricas
- ✓ UI profesional con dashboard
- ✓ ML RandomForest + Knowledge Base
- ✓ 20+ REST API endpoints
- ✓ Empaquetado .exe e instalador
- ✓ Arquitectura modular escalable

■ Próximo (v1.2.0)

- LSTM para secuencias de precios
- Reinforcement Learning agent
- Ollama/DeepSeek integration
- Ensemble models (RF + LSTM + RL)
- Mobile app (React Native)

13. Conclusión

Trading Phantom es una plataforma completa y profesional que demuestra cómo construir un bot de trading moderno con inteligencia artificial.

El proyecto está diseñado para ser:

- **Escalable:** Arquitectura modular lista para LSTM, RL, Transformers
- **Profesional:** UI moderna, API REST, backtesting visual
- **Educativo:** Código bien documentado y estructurado
- **Operacional:** Ready para producción con empaquetado profesional

Licencia: MIT (libre para usar, modificar y distribuir)

GitHub: https://github.com/blaspingto5/trading_phantom