

Botun temel amacı, kullanıcı tanımlı kâr hedeflerine ulaşırken sermaye kaybını minimumda tutmaktadır. Çoklu zaman dilimlerinde (H1 ve H4) sinyal üretimi, dinamik lot büyüklüğü hesaplama, volatiliteye uyum ve sıkı risk yönetimi kurallarıyla çalışır.

## 1.1. Ana Özellikler

- **Teknik Analiz:** RSI, MACD, Stokastik Osilatör, ATR ve Hareketli Ortalama göstergeleriyle sinyal üretimi.
- **Çoklu Zaman Dilimi:** H1 (saatlik) ve H4 (4 saatlik) zaman dilimlerinde sinyal doğrulama.
- **Risk Yönetimi:** Günlük/toplam kayıp limitleri, maksimum equity düşüşü, pozisyon korelasyon kontrolü.
- **Dinamik Mekanizmalar:** Volatiliteye bağlı stop-loss, ardışık kazançlara bağlı lot ölçeklendirme.
- **Kullanıcı Özelleştirme:** Çok sayıda parametre ile strateji esnekliği.
- **Performans Takibi:** Gerçek zamanlı durum paneli, detaylı loglama ve istatistik raporlama.
- **Ticaret Kısıtlamaları:** Spread limiti, işlem saati kısıtlamaları ve volatilite filtreleri.

## 1.2. Kod Yapısı

Bot, C# dilinde yazılmıştır ve cAlgo API'sini kullanır. Kod, aşağıdaki temel bölümlerden oluşur:

- **Parametre Tanımları:** Kullanıcı tarafından özelleştirilebilir ayarlar.
- **Göstergeler:** Teknik analiz için başlatılan göstergeler (OnStart).
- **Ana Döngü:** Her yeni bar'da çalışan işlem mantığı (OnBar).
- **Yardımcı Fonksiyonlar:** Pozisyon açma, kapatma, risk yönetimi ve performans takibi.
- **Durum Güncellemeleri:** Loglama ve durum paneli yönetimi.

## 2. Kullanıcı Parametreleri

Bot, kullanıcıların stratejiyi piyasa koşullarına ve risk iştahlarına göre özelleştirmesine olanak tanır. Aşağıda tüm parametreler, varsayılan değerleri, aralıkları ve işlevleri detaylı bir şekilde listelenmiştir:

### BOT AYAR PANELİ (TF: h1)

	Başlangıç Bakiyesi (\$):	25000.3
	Günlük Kazanç Hedefi (\$):	249.6
	Kâr Hedefi (\$):	2499.5
	Risk Yüzdesi (%):	0.7
	Take Profit (Pips):	41
	Kısmi Take Profit (Pips):	15
	Stop Loss (Pips):	23
	Günlük Kayıp Limiti (%):	1.1

 Toplam Kayıp Limiti (%): 5.5  
 Maksimum Equity Drawdown (%): 2.7

 Ticaret Başlangıç Saati (CEST): 9  
 Ticaret Bitiş Saati (CEST): 21  
 Maksimum Spread (Pips): 2

	RSI Dönemi:	9
	MACD Uzun Dönem:	29
	MACD Kısa Dönem:	13
	MACD Sinyal Dönemi:	10
	Stokastik K Dönemi:	19
	Stokastik D Dönemi:	4
	Stokastik Yavaşlatma:	7
	ATR Dönemi:	19

 Volatilite Filtresi Kullan: Hayır  
 Trend Filtresi Kullan: Hayır  
 Ticaret Saatlerini Kısıtlı: Hayır  
 Detaylı Loglama: Hayır

### Parametrelerin Stratejiye Etkisi

- RiskPercent:** Düşük değerler daha küçük lotlarla işlem açar, yüksek değerler ise riski artırır.
- TakeProfitPips ve PartialTakeProfitPips:** Kısa mesafeli take-profit, erken kâr realizasyonunu teşvik eder; uzun mesafeli take-profit, trendden daha fazla kazanç hedefler.
- UseTrendFilter:** Aktif olduğunda, işlemler yalnızca trend yönünde açılır, bu da yanlış sinyalleri azaltabilir ancak işlem sıklığını düşürebilir.
- RestrictTradingHours:** Yüksek volatilite veya likidite dönemlerine odaklanmayı sağlar (örn. Avrupa ve ABD seansları).

## 3. Teknik Göstergeler ve Sinyal Üretimi

Bot, sinyal üretmek için beş teknik göstergeyi birleştirir ve H1 ile H4 zaman dilimlerinde doğrulama yapar. Aşağıda her gösterge, parametreleri, eşikleri ve stratejideki rolü ayrıntılı olarak açıklanmıştır.

### 3.1. Kullanılan Göstergeler

#### 1. Relative Strength Index (RSI):

- Tanım:** Fiyat hareketlerinin hızını ve gücünü ölçen bir momentum göstergesi.
- Parametre:** RsiPeriod = 14.
- Zaman Dilimi:** H1 (\_rsiH1) ve M1 (\_rsi).
- Eşikler:**
  - Alım: RSI < \_rsiBuyThreshold (45).
  - Satım: RSI > \_rsiSellThreshold (55).
- Rol:** H1 zaman diliminde kısa vadeli aşırı alım/satım koşullarını belirler. M1 RSI, ek doğrulama için kullanılabilir ancak mevcut kodda

aktif değil.

- **Kod Bağlantısı:** \_rsiH1 =  
Indicators.RelativeStrengthIndex(\_h1Bars.ClosePrices, RsiPeriod).

## 2. Stochastic Oscillator:

- **Tanım:** Fiyatın kapanış seviyesini belirli bir dönemdeki fiyat aralığıyla karşılaştırın bir momentum göstergesi.
- **Parametreler:** StochasticKPeriod = 14, StochasticDPeriod = 3, StochasticSlowing = 3.
- **Zaman Dilimi:** H1 (\_stochasticH1) ve M1 (\_stochastic).
- **Eşikler:**
  - ◆ Alım: %K < \_stochasticBuyThreshold (35).
  - ◆ Satım: %K > \_stochasticSellThreshold (65).
- **Rol:** RSI ile birlikte H1'de kısa vadeli momentumu doğrular. %K hattı, hızlı fiyat hareketlerini yakalar.
- **Kod Bağlantısı:** \_stochasticH1 =  
Indicators.StochasticOscillator(\_h1Bars, ...).

## 3. MACD (Moving Average Convergence Divergence):

- **Tanım:** Trend takibi ve momentum göstergesi; kısa ve uzun vadeli EMA'lar arasındaki farkı ölçer.
- **Parametreler:** MacdLongPeriod = 26, MacdShortPeriod = 12, MacdSignalPeriod = 9.
- **Zaman Dilimi:** H4 (M1 üzerinden hesaplanır, \_macd).
- **Koşullar:**
  - ◆ Alım: MACD > 0.
  - ◆ Satım: MACD < 0.
- **Rol:** H4'te uzun vadeli trend yönünü belirler, yanlış sinyalleri filtreler.
- **Kod Bağlantısı:** \_macd = Indicators.MacdCrossOver(...).

## 4. Average True Range (ATR):

- **Tanım:** Piyasa volatilitesini ölçer; gerçek fiyat aralığının hareketli ortalaması.
- **Parametre:** AtrPeriod = 14.
- **Zaman Dilimi:** M1 (\_atr).
- **Rol:**
  - ◆ Volatilite filtresi: İşlem açmadan önce ATR'nin uygun aralıkta olduğunu kontrol eder.
  - ◆ Stop-loss hesaplama: Dinamik stop-loss için kullanılır (atrAdjustedSL).
  - ◆ Lot büyüğlüğü: Risk miktarını volatiliteye göre ayarlar.
- **Kod Bağlantısı:** \_atr = Indicators.AverageTrueRange(...).

## 5. Moving Average (MA):

- **Tanım:** Fiyatların 20 dönemlik basit hareketli ortalaması.
- **Parametre:** 20 dönem, basit MA (\_ma20).
- **Zaman Dilimi:** M1.

- **Koşullar:**
  - ◆ Alım: Kapanış fiyatı > MA20.
  - ◆ Satım: Kapanış fiyatı < MA20.
- **Rol:** UseTrendFilter aktifse, trend yönünü doğrulayarak trend karşıtı işlemleri engeller.
- **Kod Bağlantısı:** \_ma20 = Indicators.MovingAverage(...).

### 3.2. Sinyal Üretimi Mantığı

Bot, sinyal üretmek için H1 ve H4 zaman dilimlerini birleştirir. Sinyal oluşumu aşağıdaki koşullara bağlıdır:

#### H1 Koşulları (Kısa Vadeli Momentum):

- **Alım Sinyali:**
  - RSI\_H1 < 45 (aşırı satım bölgesinde).
  - Stochastic %K\_H1 < 35 (aşırı satım bölgesinde).
- **Satım Sinyali:**
  - RSI\_H1 > 55 (aşırı alım bölgesinde).
  - Stochastic %K\_H1 > 65 (aşırı alım bölgesinde).

#### H4 Koşulları (Uzun Vadeli Trend):

- **Alım Sinyali:**
  - MACD > 0 (yükseliş trendi).
  - Eğer UseTrendFilter = true ise: Kapanış Fiyatı > MA20.
- **Satım Sinyali:**
  - MACD < 0 (düşüş trendi).
  - Eğer UseTrendFilter = true ise: Kapanış Fiyatı < MA20.

#### Sinyal Onayı:

- Hem H1 hem de H4 koşulları sağlanırsa, bot işlem açar (OpenPosition).
- Örnek:
  - Alım: RSI\_H1 < 45 && Stochastic %K\_H1 < 35 && MACD > 0 && (UseTrendFilter ? Price > MA20 : true).
  - Satım: RSI\_H1 > 55 && Stochastic %K\_H1 > 65 && MACD < 0 && (UseTrendFilter ? Price < MA20 : true).

### 3.3. Eşiklerin Dinamik Ayarlanması

- **Mevcut Durum:** AdjustDynamicThresholds fonksiyonu, RSI ve Stokastik eşiklerini sabit tutar (\_rsiBuyThreshold = 45, \_rsiSellThreshold = 55, \_stochasticBuyThreshold = 35, \_stochasticSellThreshold = 65).
- **Potansiyel Geliştirme:** Kod, dinamik eşik ayarlaması için bir çerçeve sunar, ancak bu özellik şu an kullanılmamaktadır. Gelecekte, piyasa koşullarına (örn. volatilite, trend gücü) göre eşikler otomatik olarak ayarlanabilir.

## 4. İşlem Açıma Stratejisi

Bot, sinyal alındığında iki ayrı pozisyon açar: **kısmi take-profit** ve **tam take-profit** pozisyonları. Bu strateji, erken kâr realizasyonu ile uzun vadeli kazanç potansiyelini birleştirir.

### 4.1. İşlem Açıma Koşulları

Bot, işlem açmadan önce bir dizi kontrol yapar:

#### 1. Spread Kontrolü:

- currentSpread = Symbol.Spread / Symbol.PipSize.
- Eğer currentSpread > MaxSpreadPips (3.0 pip), işlem açılmaz.
- Kod: if (currentSpread > MaxSpreadPips) { Print("Spread çok yüksek..."); return; }.

#### 2. Pozisyon Korelasyon Kontrolü:

- Aynı yönde (TradeType) ve sembolde açık pozisyon varsa yeni işlem açılmaz (CheckPositionCorrelation).
- Kod: if (!CheckPositionCorrelation(tradeType)) return;.

#### 3. Maksimum Pozisyon Sınırı:

- Aynı anda en fazla 2 pozisyon açık olabilir (Positions.Count >= 2).
- Kod: if (Positions.Count >= 2) { Print("Maksimum açık pozisyon..."); return; }.

#### 4. Volatilite Filtresi:

- Eğer UseVolatilityFilter = true, ATR değeri uygun aralıkta olmalıdır:
  - ♦ dynamicMin = Symbol.PipSize \* 0.75.
  - ♦ dynamicMax = Symbol.PipSize \* 20 \* (spread > \_averageSpread ? 1.5 : 1).
  - ♦ Kod: if (UseVolatilityFilter && !IsVolatilitySuitable()) { Print("Volatilite uygun değil..."); return; }.

#### 5. Ticaret Saati Kısıtlaması:

- Eğer RestrictTradingHours = true, işlemler sadece TradingStartTime (06:00) ile TradingEndTime (23:00) arasında açılır.
- Kod: if (currentHour < TradingStartTime || currentHour >= TradingEndTime) { Print("Ticaret zaman aralığı dışında..."); return; }.

### 4.2. Lot Büyüklüğü Hesaplama

Lot büyüğü, risk yönetimi ve volatiliteye bağlı olarak dinamik şekilde hesaplanır:

- Risk Tabanlı Hesaplama:

- riskAmount = Account.Balance \* (RiskPercent / 100) (örn. %0.5).
- atrAdjustedSL = Math.Max(StopLossPips, \_atr.Result.LastValue / Symbol.PipSize \* 1.5).
- lotSize = riskAmount / (atrAdjustedSL \* Symbol.PipValue).

- Maksimum Lot Sınırı:

- maxLotSize = (\_startingTotalBalance \* 0.0075) / (atrAdjustedSL \* Symbol.PipValue).
- lotSize = Math.Min(lotSize, maxLotSize).

- Dinamik Lot Ölçeklendirme:

- Eğer \_consecutiveWins >= 5 (5 ardışık kazanç), lot büyüğü %25 artırılır:
  - ♦ scaledLot = lotSize × 1.25.
  - ♦ lotSize = Math.Min(Symbol.NormalizeVolumeInUnits(scaledLot,

RoundingMode.Down), maxLotSize).

- **Normalizasyon:**

- lotSize = Symbol.NormalizeVolumeInUnits(lotSize, RoundingMode.Down) ile sembole uygun lot birimi kullanılır.

- **Sonuç:** Toplam lot, iki eşit parçaya bölünür (halfLot = lotSize / 2).

### 4.3. Pozisyon Türleri

#### 1. Kısmi Take-Profit Pozisyonu:

- **Lot:** halfLot.
- **Stop-Loss:** atrAdjustedSL.
- **Take-Profit:** PartialTakeProfitPips (20 pip).
- **Etiket:** FTMO\_Bot\_Partial.
- **Amaç:** Erken kâr realizasyonu için kısa mesafeli hedef.
- **Kod:** ExecuteMarketOrder(tradeType, SymbolName, halfLot, "FTMO\_Bot\_Partial", (int)atrAdjustedSL, PartialTakeProfitPips).

#### 2. Tam Take-Profit Pozisyonu:

- **Lot:** halfLot.
- **Stop-Loss:** atrAdjustedSL.
- **Take-Profit:** TakeProfitPips (60 pip).
- **Etiket:** FTMO\_Bot\_Full.
- **Amaç:** Daha uzun vadeli trendden kâr elde etmek.
- **Kod:** ExecuteMarketOrder(tradeType, SymbolName, halfLot, "FTMO\_Bot\_Full", (int)atrAdjustedSL, TakeProfitPips).

### 4.4. Stop-Loss ve Take-Profit

- **Stop-Loss:**

- Dinamik olarak hesaplanır: atrAdjustedSL = Max(StopLossPips, ATR / PipSize \* 1.5).
- Minimum StopLossPips = 75 pip.
- Volatiliteye bağlı olarak stop-loss genişletilir, böylece ani fiyat hareketlerinden korunur.

- **Take-Profit:**

- Kısmi pozisyon: PartialTakeProfitPips = 20 pip (hızlı kâr realizasyonu).
- Tam pozisyon: TakeProfitPips = 60 pip (daha büyük hareketleri yakalama).

- **Trailing Stop:**

- Tam pozisyonlarda (FTMO\_Bot\_Full), pozisyon 15 pip kârda olduğunda stop-loss, giriş fiyatının 10 pip ötesine taşınır:
  - ◆ Alım: newSL = position.EntryPrice + (10 \* Symbol.PipSize).
  - ◆ Satım: newSL = position.EntryPrice - (10 \* Symbol.PipSize).
  - ◆ Kod: ModifyPosition(position, newSL, position.TakeProfit).

### 4.5. İşlem Açma Örneği

- **Scenario:** Alım sinyali alındı, Account.Balance = \$25,000, RiskPercent = 0.5, ATR = 0.0015, PipValue = 10.

- **Hesaplama:**

- riskAmount =  $25,000 * 0.005 = \$125$ .
- atrAdjustedSL =  $\text{Max}(75, 0.0015 / 0.0001 * 1.5) = \text{Max}(75, 22.5) = 75 \text{ pip}$ .
- lotSize =  $125 / (75 * 10) = 0.1667 \text{ lot}$ .
- maxLotSize =  $(25,000 * 0.0075) / (75 * 10) = 0.25 \text{ lot}$ .
- lotSize =  $\text{Min}(0.1667, 0.25) = 0.16 \text{ lot} (\text{normalleştirilmiş})$ .
- halfLot = 0.08 lot.

- **Pozisyonlar:**

- K ruling: FTMO\_Bot\_Partial: 0.08 lot, SL: 75 pip, TP: 20 pip.
- FTMO\_Bot\_Full: 0.08 lot, SL: 75 pip, TP: 60 pip.

## 5. Risk Yönetimi

Bot, sermaye korumasını sağlamak için çok katmanlı risk yönetimi kuralları uygular.

### 5.1. Kayıp Limitleri

#### 1. Günlük Kayıp Limiti:

- maxDailyLossLimit =  $_{\text{startingDailyBalance}} * (\text{DailyLossLimitPercent} / 100)$  (varsayılan: %0.25).
- Hesaplama: currentDailyLoss =  $(_{\text{startingDailyBalance}} - \text{Account.Balance}) - (\text{Account.Equity} - \text{Account.Balance})$ .
- Eğer currentDailyLoss  $\geq$  maxDailyLossLimit:
  - ◆ Yeni işlem açılmaz.
  - ◆ Zarar eden pozisyonlar kapatılır (ReducePositions(false)).
- Kod: if (currentDailyLoss  $\geq$  maxDailyLossLimit) { Print("Günlük Maksimum Kayıp Limiti..."); ReducePositions(false); return; }.

#### 2. Toplam Kayıp Limiti:

- maxTotalLossLimit =  $_{\text{startingTotalBalance}} * (\text{TotalLossLimitPercent} / 100)$  (varsayılan: %0.5).
- Hesaplama: currentTotalLoss =  $_{\text{startingTotalBalance}} - \text{Account.Equity}$ .
- Eğer currentTotalLoss  $\geq$  maxTotalLossLimit:
  - ◆ Tüm pozisyonlar kapatılır (ReducePositions(true)).
  - ◆ Bot durur (Stop()).
- Kod: if (currentTotalLoss  $\geq$  maxTotalLossLimit) { Print("Toplam Maksimum Kayıp Limiti..."); ReducePositions(true); Stop(); return; }.

#### 3. Pozisyon Bazlı Kayıp Limiti:

- Her pozisyonun maksimum kaybı:
  - ◆ tradeMaxLoss =  $\text{StartingBalance} * 0.05 / 20$  (örn. \$25,000 için \$62.5).
  - ◆ Veya Account.Balance \* 0.005 (%0.5).
- Eğer position.NetProfit < -tradeMaxLoss veya position.NetProfit < -Account.Balance \* 0.005:
  - ◆ Pozisyon kapatılır.
  - ◆ Kod: if (position.NetProfit < -tradeMaxLoss)

{ ClosePosition(position); ... }.

## 5.2. Equity Drawdown Kontrolü

- **Maksimum Equity Drawdown:**

- currentEquityDrawdown = (\_maxEquity - Account.Equity) / \_maxEquity \* 100.
- Eğer currentEquityDrawdown >= MaxEquityDrawdownPercent (varsayılan: %3):
  - ◆ Tüm pozisyonlar kapatılır.
  - ◆ Bot durur.
- Kod: if (currentEquityDrawdown >= MaxEquityDrawdownPercent) { Print("Maksimum Equity Drawdown..."); ReducePositions(true); Stop(); return; }.

- **NAV Sapma Kontrolü:**

- navDeviation = (Account.Balance - Account.Equity) / Account.Balance \* 100.
- Eğer navDeviation >= 1.0:
  - ◆ Tüm pozisyonlar kapatılır.
- Kod: if (navDeviation >= 1.0) { Print("NAV sapma limiti..."); ReducePositions(true); return; }.

## 5.3. Kâr Hedefi Kontrolü

- Eğer Account.Balance - StartingBalance >= ProfitTarget (varsayılan: \$2500):
  - Bot durur.
- Kod: if (Account.Balance - StartingBalance >= ProfitTarget) { Print("Kâr hedefi..."); Stop(); return; }.

## 6. Pozisyon Yönetimi

Bot, açık pozisyonları dinamik olarak yönetir.

### 6.1. Trailing Stop

- **Koşul:** Tam pozisyonlarda (FTMO\_Bot\_Full), pozisyon 15 pip kârda olduğunda:
  - Alım: newSL = position.EntryPrice + (10 \* Symbol.PipSize).
  - Satım: newSL = position.EntryPrice - (10 \* Symbol.PipSize).
- **Kontrol:** Yeni stop-loss, mevcut fiyattan en az 2 pip uzakta olmalıdır.

### 6.2. Pozisyon Kapatma

- **Zarar Eden Pozisyonlar:** ReducePositions(false) ile zarar eden pozisyonlar kapatılır (günlük kayıp limiti aşıldığında).
- **Tüm Pozisyonlar:** ReducePositions(true) ile tüm pozisyonlar kapatılır (toplam kayıp, equity drawdown veya NAV sapma limitleri aşıldığında).

## 7. Performans Takibi ve Loglama

Bot, performansını izlemek ve kullanıcıya bilgi sağlamak için gelişmiş araçlar sunar.

### 7.1. Durum Paneli

- **İçerik:**

- Bakiye (Account.Balance).
  - Equity (Account.Equity).
  - Toplam kâr (\_totalProfit).
  - Ticaret günü sayısı (\_tradingDays.Count).
  - Toplam işlem sayısı (\_totalTrades).
  - **Görsel:** Chart üzerinde statik metin olarak gösterilir (Chart.DrawStaticText).
  - **Kod:**
- 7.2. Detaylı Loglama**
- Eğer DetailedLogging = true:
    - Her bar için: Print("Durum: Equity=", Account.Equity, ", Balance=", Account.Balance, ", Pozisyonlar=", Positions.Count);.
  - Sinyal bilgileri: Print("Alım sinyali: RSI\_H1=", \_rsiH1.Result.LastValue, ...);.
  - Hata mesajları: Print("İşlem açılamadı: ", result1.Error?.ToString() ?? result2.Error?.ToString());.

### 7.3. İstatistik Raporlama

- Bot durdurulduğunda (OnStop):
  - Toplam işlem sayısı (\_totalTrades).
  - Kazanma oranı: winRate = (\_winningTrades / \_totalTrades) \* 100.
  - Toplam kâr (\_totalProfit).
  - Net bakiye (Account.Balance).
  - Net kâr (Account.Balance - StartingBalance).
  - Son eşikler: RSI (\_rsiBuyThreshold, \_rsiSellThreshold), Stokastik (\_stochasticBuyThreshold, \_stochasticSellThreshold).
- **Minimum Ticaret Günü Uyarısı:**
  - Eğer \_tradingDays.Count < 4: Print("UYARI: Minimum 4 ticaret günü sağlanmadı...");.
- 

## 8. Kod Analizi: Kritik Fonksiyonlar

Aşağıda botun temel fonksiyonları ve işlevleri ayrıntılı olarak açıklanmıştır.

### 8.1. OnStart

- **Amaç:** Botun başlangıcında göstergeleri ve değişkenleri başlatır.
- **İşlemler:**
  - Göstergeler başlatılır: \_rsi, \_macd, \_stochastic, \_atr, \_ma20, \_rsiH1, \_stochasticH1.
  - H1 barları yüklenir: \_h1Bars = MarketData.GetBars(TimeFrame.Hour).
  - Başlangıç değerleri atanır: \_startingTotalBalance, \_startingDailyBalance, \_currentDay, \_maxEquity, \_averageSpread.
  - Hata yönetimi: Göstergeler başlatılamazsa bot durur (Stop()).
- 

### 8.2. OnBar

- **Amaç:** Her yeni bar'da işlem mantığını çalıştırır.

- **Adımlar:**

1. Günlük bakiye güncellenir (yeni günse).
2. İşlem saati kontrolü.
3. Kayıp limitleri kontrolü (günlük, toplam, equity, NAV).
4. Kâr hedefi kontrolü.
5. Pozisyon sayısı kontrolü.
6. Volatilite filtresi.
7. Sinyal üretimi ve işlem açma.
8. Pozisyon yönetimi.
9. Durum paneli güncelleme.

### 8.3. OpenPosition

- **Amaç:** Alım veya satım pozisyonu açar.

- **Adımlar:**

1. Spread kontrolü.
2. Pozisyon korelasyon kontrolü.
3. Lot büyüklüğü hesaplama.
4. İki pozisyon açma: Kısmi ve tam take-profit.
5. İşlem sonuçlarının loglanması.

•

### 8.4. ManagePositions

- **Amaç:** Açık pozisyonları izler ve yönetir.

- **Adımlar:**

1. Trailing stop uygular.
2. Pozisyon bazlı kayıp limitlerini kontrol eder.

## 9. Avantajlar ve Sınırlamalar

### 9.1. Avantajlar

1. **Çoklu Zaman Dilimi Analizi:** H1 ve H4 kombinasyonu, sinyal doğruluğunu artırır.
2. **Gelişmiş Risk Yönetimi:** Çok katmanlı kayıp limitleri, trailing stop ve dinamik stop-loss.
3. **Dinamik Mekanizmalar:** Volatiliteye bağlı stop-loss, ardışık kazançlara bağlı lot ölçeklendirme.
4. **Kullanıcı Dostu:** Özelleştirilebilir parametreler, durum paneli ve detaylı loglama.
5. **Performans Takibi:** Gerçek zamanlı istatistikler ve kapsamlı durdurma raporu.

### 9.2. Sınırlamalar

1. **Sabit Eşikler:** RSI ve Stokastik eşikleri dinamik olarak ayarlanmıyor, bu da bazı piyasa koşullarında performansı sınırlayabilir.
2. **Tek Sembol Desteği:** Bot, yalnızca tek bir simbol üzerinde çalışır; çoklu simbol desteği yoktur.
3. **Kompleksite:** Çok sayıda parametre, yeni kullanıcılar için öğrenme eğrisini artırabilir.
4. **Volatilite Bağımlılığı:** Düşük volatilite veya yüksek spread

dönemlerinde işlem sıklığı azalabilir.

5. **Hata Yönetimi:** Bazı hata senaryoları (örn. bağlantı kesilmesi) için ek kontroller eksik.

## 10. Potansiyel İyileştirme Alanları

1. **Dinamik Eşikler:** AdjustDynamicThresholds fonksiyonuna volatilite veya trend gücüne bağlı eşik ayarlaması eklenebilir.
2. **Çoklu Sembol Desteği:** Farklı sembollerde eşzamanlı işlem yapma yeteneği eklenebilir.
3. **Gelişmiş Volatilite Filtresi:** ATR yerine VIX veya diğer volatilite göstergeleri kullanılabilir.
4. **Ek Risk Kontrolleri:** Maksimum açık pozisyon sayısı dinamik olarak ayarlanabilir.
5. **Backtest Optimizasyonu:** Parametrelerin geçmiş verilerle optimize edilmesi için bir çerçeve eklenebilir.
6. **Hata Dayanıklılığı:** Bağlantı sorunları veya API hataları için yeniden deneme mekanizmaları eklenebilir.

## 11. Örnek Senaryo: Botun Çalışma Mantığı

- **Başlangıç Koşulları:**
  - Bakiye: \$25,000.
  - Sembol: EUR/USD.
  - Risk: %0.5.
  - ATR: 0.0015 (15 pip).
  - Spread: 1.5 pip.
- **Sinyal:**
  - H1: RSI\_H1 = 40, Stochastic %K\_H1 = 30.
  - H4: MACD > 0, Price > MA20.
- **İşlem:**
  - Lot: 0.16 lot (halfLot = 0.08).
  - Kısmi pozisyon: SL: 75 pip, TP: 20 pip.
  - Tam pozisyon: SL: 75 pip, TP: 60 pip.
- **Yönetim:**
  - Tam pozisyon 15 pip kârda: SL, giriş + 10 pip'e taşınır.
  - Kısmi pozisyon 20 pip'te kapanır: Kâr = 0.08 \* 20 \* 10 = \$16.
- **Sonuç:**
  - Kâr hedefi (\$2500) veya kayıp limiti (%0.5) aşılırsa bot durur.