تقرير تقييم شامل

# نظام التداول الآلي
# رؤى ترايدنج

فحص وتقييم حالة تنفيذ المراحل الأربع

الإصلاحات - تصحيح البيانات - التحسين الخوارزمي - التطوير الهيكلي

تاريخ التقرير: 9 يونيو 2026

إعداد: Z.ai | الإصدار: 1.0

# فهرس المحتويات


| | |
|---|---|
| الملخص التنفيذي | 2 |
| المرحلة الأولى: الإصلاح الطارئ (8 أخطاء) | 2 |
| المرحلة الثانية: تصحيح البيانات (4 شذوذ) | 4 |
| المرحلة الثالثة: التحسين الخوارزمي (5 تحسينات) | 5 |
| المرحلة الرابعة: التطوير الهيكلي (4 تطويرات) | 7 |
| الفجوات الحرجة والمخاطر | 9 |
| الخلاصة والتوصيات | 10 |


# الملخص التنفيذي

**67%**

**مستوى التنفيذ الإجمالي**

من أصل 21 عنصرًا مطلوبًا عبر المراحل الأربع:

13 مُنفَّذ بالكامل | 3 مُنفَّذ جزئيًا | 5 غير مُنفَّذ

| المرحلة 1 | المرحلة 2 | المرحلة 3 | المرحلة 4 |
|---|---|---|---|
| **8/8** | **4/4** | **4/5** | **1/4** |
| الإصلاح الطارئ | تصحيح البيانات | التحسين الخوارزمي | التطوير الهيكلي |

**تحذير حرج:**

المرحلتان 3 و4 تحتويان على فجوات تنفيذية مهمة. تحيز اتجاه البيع (#14) مُعالَج جزئيًا فقط، والمرحلة الرابعة بأكملها (توحيد المسارات، تقسيم AIOrchestrator، توحيد المؤشرات) لم تُنفَّذ بعد باستثناء نظام رصد الأداء.

يتضمن هذا التقرير فحصًا معمقًا لكل عنصر من عناصر المراحل الأربع، مع تقييم حالة التنفيذ وجودته والفجوات المتبقية والمخاطر المحتملة. تم الفحص بناءً على قراءة الكود المصدري الفعلي لجميع الخدمات المعنية في المشروع.

# المرحلة الأولى: الإصلاح الطارئ

الأسبوع 1-2 | 8 أخطاء حرجة وعالية | مكتمل 100%

| # | الإصلاح | الحالة | الملف الرئيسي | الجودة |
|---|---|---|---|---|
| 1 | هامش Paper Trading لا يُخصم فعليًا | مُنفَّذ | trading.service.ts | قوي - خصم ذري via $executeRaw |
| 2 | Alpaca يعمل دائمًا في وضع Paper | مُنفَّذ | execution-gateway.service.ts | قوي - متغير بيئة + علم credentials |
| 3 | أوامر Limit الورقية لا تُنفَّذ أبدًا | مُنفَّذ | paper-trading.adapter.ts | جيد - فحص كل 10 ثوانٍ + إلغاء تلقائي 24h |
| 4 | تسجيل BullMQ مكرر يسبب انهيار | جزئي | order-queue.processor.ts | تحذير فقط - المكرر لا يزال يسجَّل |
| 5 | تخزين مفاتيح API مشفرة في الذاكرة المؤقتة | مُنفَّذ | execution-gateway.service.ts | قوي - TTL من 5 دقائق إلى 60 ثانية |
| 6 | فحوصات مخاطر مزدوجة في V1 | مُنفَّذ | trading.service.ts + risk-gatekeeper | قوي - skipRiskCheck flag |
| 7 | تباعد سلوكي بين V1 و V2 | مُنفَّذ | position-monitor.service.ts | قوي - STALE_POSITION + TIME_EXPIRED |
| 8 | ExchangeSync يجلب صفقات جميع المستخدمين | مُنفَّذ | exchange-sync.service.ts | قوي - التحقق من userId |

## تفاصيل الإصلاحات الرئيسية

### الإصلاح #1: خصم الهامش الذري

تم نقل خصم الهامش من التطبيق إلى قاعدة البيانات باستخدام `executeRaw$` مع أمر SQL ذري: `SET "paperBalance" = "paperBalance" - ${marginToDeduct}`. هذا يمنع سباق التزامن عند معالجة أوامر متزامنة تقرأ نفس الرصيد. يتم حساب الهامش مع مراعاة الرافعة المالية (فوركس 50x، ذهب 20x، كريبتو 1x افتراضي). كما تم إضافة تحقق من السعر يرفض الأوامر بسعر ومنعتسجيلصفقاتوهمية بقيمة 0، ومنعتسجيلصفقاتوهمية بقيمة 0، 0.00.

### الإصلاح #2: تبديل Alpaca Paper/Live

يعتمد النظام الآن على متغير البيئة `ALPACA_LIVE_ENABLED` وعلم `isCredentialTestnet` لتحديد الوضع. الوضع الافتراضي هو Paper. يتطلب التبديل إلى Live موافقة صريحة مع تسجيل تحذير أمني وتدقيق في سجل المراجعة. تم أيضًا إصلاح مشكلة حرجة: قاعدة البيانات تخزّن `'paper-trading'` لكن التبديل كان يطابق فقط `'paper'`، مما كان يتسبب في فشل جميع أوامر Paper Trading حيث كانت تُمرَّر لـ BinanceAdapter بمفاتيح وهمية.

### الإصلاح #3: أوامر Limit الورقية

تمت إضافة فاحص دوري `_checkPendingLimitOrders()` يعمل كل 10 ثوانٍ. يقارن سعر السوق بسعر الأمر المحدد ويملأ الأمر عند تحقق الشروط (BUY: سعر السوق أقل أو يساوي سعر الأمر، SELL: العكس). الأوامر التي تتجاوز 24 ساعة تُلغى تلقائيًا. تم أيضًا إضافة دالة `destroy()` لتنظيف المؤقت عند انتهاء صلاحية الـ Adapter في الذاكرة المؤقتة.

**تنبيه - الإصلاح #4 (BullMQ Singleton):**

الحارس الحالي يسجّل تحذيرًا فقط عند اكتشاف معالج مكرر، لكن المكرر لا يزال يسجّل نفسه في BullMQ ويعالج المهام. الإصلاح الفعلي يتطلب منع الاستنساخ الثاني من الاتصال بـ BullMQ تمامًا، إما برمي استثناء في المنشئ أو استخدام حارس على مستوى الوحدة النمطية.

**تنبيه - نافذة سباق خصم الهامش:**

الخصم يحدث بعد المعاملة التي تنشئ الأمر والمركز، وليس داخلها. إذا تعطلت الخدمة بين المعاملة والخصم، يبقى المركز مفتوحًا بدون خصم هامش. يُنصح بنقل `executeRaw$` داخل المعاملة.

**تنبيه - أوامر Limit عند انتهاء صلاحية Adapter:**

كل نسخة PaperTradingAdapter تنشئ `setInterval` خاص بها. عند انتهاء صلاحية Adapter في الذاكرة المؤقتة (TTL 60 ثانية) وتحويله لـ `destroy()`، تُفقد أوامر Limit المعلقة في Map الداخلي. مفاتيح Redis تبقى لكن لا شيء يفحصها بعد تدمير Adapter القديم.

# المرحلة الثانية: تصحيح البيانات

الأسبوع 2-3 | 4 شذوذ حرجة في البيانات | مكتمل 100%

| # | الشذوذ | الحالة | آلية المعالجة | ملاحظة |
|---|---|---|---|---|
| 9 | سعر BTC مسجل بـ 1,921 بدلاً من 1,921 بدلاً من 63,000 | مُنفَّذ | PRICE_SANITY متعدد الطبقات | النطاقات ثابتة ومحدودة |
| 10 | صفقة ETH عالقة 131 ساعة | مُنفَّذ | STALE_POSITION + TIME_EXPIRED | الكاشف يغطي Paper فقط |
| 11 | صفقات DOGE/SOL متكررة كل 8-10 ثوانٍ | مُنفَّذ | Cooldown + فلتر التكرار | قفل الاتجاه قد لا يعمل |
| 12 | نمط إغلاق "يدوي" بعد 4 ساعات | مُنفَّذ | closeReason افتراضي = AUTO_CLOSE | البيانات القديمة غير مصححة |

## تفاصيل التصحيحات

### التصحيح #9: التحقق من صحة الأسعار (4 طبقات)

تم إنشاء خريطة `PRICE_SANITY` ثابتة تحدد نطاقات سعرية لكل رمز (مثلاً BTC/USDT: 20,000 − −000 ,20 200,000). تُطبَّق على أربع طبقات مستقلة: خدمة التبادل (exchange.service.ts)، خدمة التجميع (aggregator.service.ts)، المنفذ الذكي (smart-executor.service.ts)، والمجلس الاستراتيجي (strategic-council.service.ts). إذا خرج السعر عن النطاق في طبقة ما، يُنتقل إلى مصدر بديل. إذا فشلت جميع المصادر، يُعاد سعر 0 مما يمنع التداول بشكل آمن. هناك أيضًا طبقة خامسة في خدمة بيانات السوق (market-data.service.ts) تعمل كفلتر أولي قبل التجميع.

> **فجوة:**
> النطاقات ثابتة ولا تشمل جميع الرموز. فقط ~10 رموز لها نطاقات تحقق. أي رمز غير مُدرَج يمر بدون تحقق. كما أن الحد الأدنى لـ BTC ($20,000) سيصبح قديمًا إذا انخفض السعر عن هذا المستوى. لا توجد آلية تحديث تلقائي للنطاقات.

### التصحيح #10: الصفقات العالقة

تمت إضافة كاشف الصفقات الراكدة الذي يفحص كل 30 ثانية. مراكز Paper Trading المفتوحة لأكثر من 48 ساعة بدون SL/TP تُغلق تلقائيًا بسبب `STALE_POSITION` . بالإضافة إلى ذلك، `MAX_HOLDING_TIME` يحدد أقصى مدة احتفاظ حسب الإطار الزمني: M1/M5 = 4 ساعات، M15/M30 = 12 ساعة، H1/H4 = 48 ساعة، D1+ = 7-14 يوم. المنفذ الذكي أيضًا يغلق تلقائيًا المراكز الورقية الراكدة التي تتجاوز 24 ساعة عند بدء التشغيل.

> **فجوة:**
> كاشف الصفقات الراكدة يغطي فقط بورصة Paper Trading. المراكز على البورصات الحقيقية بدون SL/TP لن تُغلق تلقائيًا بهذا الكاشف - فقط تحذير العمر (7 أيام) يعمل. كما أن الإطار الزمني يُخزَّن في Redis، وإذا انتهت صلاحية المفتاح قبل إغلاق المركز، يُستخدَم احتياطي 8 ساعات.

## التصحيح #11: تكرار الصفقات

تم بناء دفاع متعدد الطبقات: (1) Cooldown لمدة 5 دقائق بعد إغلاق TIME_EXPIRED أو STOP_LOSS عبر مفتاح `cooldown:{userId}:{symbol}` ، (2) Redis فحص Cooldown في المنفذ الذكي قبل معالجة الإشارات يُصفّي الإشارات في فترة التهدئة، (3) فلتر تكرار الصفقات في RiskGatekeeper بثلاث قواعد: قفل الاتجاه 30 دقيقة، حد أقصى 5 صفقات/رمز/يوم، 3 خسائر متتالية = حظر ساعتين.

> **فجوة حرجة:**
> مفتاح Redis لقفل الاتجاه `trade-rep:dir-lock:{userId}:{symbol}:{side}` يتم فحصه في RiskGatekeeper (السطور 1262-1271) لكن لم يُعثر على المكان الذي يُكتَب فيه عند إغلاق المركز. القاعدتان 2 و3 (الحد اليومي والخسائر المتتالية) تعملان بشكل صحيح مع مسارات القراءة والكتابة. لكن القاعدة الأولى (قفل 30 دقيقة) قد تكون غير فعالة عمليًا بسبب عدم كتابة المفتاح.

## التصحيح #12: نمط الإغلاق اليدوي

تم تغيير القيمة الافتراضية لـ `closeReason` من `'MANUAL'` إلى `'AUTO_CLOSE'` في دالة `closePosition()` . الآن الإغلاقات التلقائية تُسجَّل بشكل صحيح بدلاً من أن تبدو وكأنها إغلاقات يدوية. الإغلاق بعد 4 ساعات للإطارات M1/M5 هو تصميم مقصود (MAX_HOLDING_TIME) وليس خطأ. البيانات التاريخية التي سُجلت كـ MANUAL قبل هذا الإصلاح لن تُصحَّح تلقائيًا - لا توجد عملية ترحيل لقاعدة البيانات.

# المرحلة الثالثة: التحسين الخوارزمي

الأسبوع 3-5 | 5 تحسينات جوهرية | مكتمل جزئيًا - 4/5

| # | التحسين | الحالة | الآلية | الفجوة الرئيسية |
|---|---|---|---|---|
| 13 | إعادة تصميم نسبة المخاطرة/المكافأة | مُنفَّذ | Minimum R:R = 1.5 (V1) + استراتيجي (Agent) | تعارض عتبات V1 و Agent |
| 14 | معالجة تحيز اتجاه البيع | جزئي | تعزيز 1.3x في المجلس الاستراتيجي | لا يُطبَّق على Agent أو SignalEvaluator |
| 15 | تحسين أحجام المراكز | مُنفَّذ | حجم قائم على المخاطرة + حدود محفظة | Paper يتجاوز فحص % المحفظة |
| 16 | تطوير آلية إغلاق ذكية | مُنفَّذ | Break-Even + Trailing Stop + إغلاق زمني | مراكز Agent مستثناة |
| 17 | فلتر تكرار الصفقات | مُنفَّذ | 3 قواعد Cooldown + Redis | فشل مفتوح عند تعطل Redis |

## تفاصيل التحسينات

### #13: نسبة المخاطرة/المكافأة (SL:TP)

تم تنفيذ نظام مزدوج: في مسار V1، يفرض RiskGatekeeper حدًا أدنى 1.5:1 كنسبة R:R عبر دالة `enforceStopLoss()`. في مسار Agent، يستخدم RiskCalculator جدول عتبات استراتيجي محدد: DCA = 0.4، Swing = 1.5، Momentum = 1.2، Scalping/VWAP = 1.0، Mean Reversion/Grid = 0.8. هذا النهج الاستراتيجي أكثر تطورًا ومراعاة لطبيعة كل استراتيجية - فاستراتيجية DCA تعتمد على معدل فوز عالٍ لتعويض نسبة R:R المنخفضة.

**تعارض العتبات:**

صفقة DCA بنسبة R:R = 0.6 ستمر عبر RiskCalculator للـ Agent لكن ستُرفَض من RiskGatekeeper عند عتبة 1.5. العتبات الاستراتيجية في Agent تُلغَى فعليًا بواسطة العتبة الأكثر صرامة في RiskGatekeeper. كما أن فحص R:R في RiskGatekeeper يعمل فقط عند توفير TakeProfit، وإذا لم يُحدَّد TP يتخطى الفحص تمامًا مما يخلق ثغرة.

### #14: تحيز اتجاه البيع (85%+ صفقات بيع) - جزئي

تم تنفيذ آلية في المجلس الاستراتيجي (Strategic Council): عند تعارض إشارات BUY و SELL، يُحسَب نسبة الاتجاهات لليوم. إذا تجاوز اتجاه واحد 70% (مع حد أدنى 4 صفقات)، يحصل الاتجاه المعاكس على تعزيز 1.3x في النقاط. كما أن فلتر تكرار الصفقات يحد بشكل غير مباشر من تحيز البيع عبر قفل الاتجاه لمدة 30 دقيقة.

**فجوة كبيرة:**

التعزيز 1.3x يُطبَّق فقط في خطوة توحيد الإشارات بالمجلس الاستراتيجي (عند تعارض BUY/SELL). لا يُطبَّق عندما تكون هناك إشارات SELL فقط (لا تعارض = لا تعزيز). الأهم: Agent لا يملك أي وعي بتحيز الاتجاه - خدمات SignalEvaluator و

AdaptiveStrategySelector تولّد إشارات بدون فحص نسبة البيع/الشراء التاريخية. لا يوجد حد أقصى صارم لنسبة البيع (مثلاً "أقصى 75% صفقات بيع يوميًا"). المعلمات (عتبة 70%، تعزيز 1.3x) ثابتة غير قابلة للتهيئة.

## #15: أحجام المراكز

تم بناء نظام متعدد الطبقات: RiskCalculator يستخدم الحجم القائم على المخاطرة ( `riskAmount = portfolio x` `%riskPerTradePercent` ) مع مراعاة أحجام اللوت (فوركس: 100K وحدة/لوت، كريبتو: متغير). الحد الأقصى 2% = `maxPositionSizePercent` من المحفظة. RiskGatekeeper يتحقق من حدود حجم المركز مع وعي بالمصدر (Executor مقابل Agent). خصم الهامش ذري مع حساب الرافعة. هناك دالة `roundLotSize()` لتقريب أحجام اللوت في أزواج الفوركس و `lotsToUnits()` للتحويل بين اللوتات والوحدات.

**فجوة:**

RiskGatekeeper يتجاوز فحص نسبة المركز من المحفظة لـ Paper Trading، ويفحص فقط عدد المراكز والهامش. هذا قد يسمح بمراكز ورقية كبيرة الحجم. كما أن الفجوة بين `riskPerTradePercent` الافتراضي (1.5%) و `maxPositionSizePercent` (2%) ضيقة مما قد يؤدي للحد من حجم المركز عند استخدام SL واسع.

## #16: آلية الإغلاق الذكية

تم تطوير نظام إغلاق ذكي شامل يتضمن أربع آليات متكاملة تعمل ضمن Position Monitor الذي يفحص المراكز كل 30 ثانية:

| الآلية | شروط التفعيل | التفاصيل |
|---|---|---|
| Break-Even Stop | ربح غير محقق +1% | نقل SL إلى نقطة الدخول + 0.01% لضمان تغطية الرسوم |
| Trailing Stop | ربح غير محقق +2% | مسافة تتبع 1.2% من أعلى سعر (BUY) أو أقل سعر (SELL) |
| إغلاق زمني | تجاوز MAX_HOLDING_TIME | M1/M5=4h، M15/M30=12h، H1/H4=48h، D1+=7-14d |
| إغلاق راكد | مركز Paper أكثر من 48h بدون SL/TP | إغلاق تلقائي بسبب STALE_POSITION |

بعد الإغلاق التلقائي (TIME_EXPIRED أو STOP_LOSS)، يُفعَّل Cooldown لمدة 5 دقائق عبر Redis لمنع إعادة فتح المركز فورًا. نظام إعادة المحاولة يستخدم `closePositionWithRetry()` مع 3 محاولات، مع احتياطي `forceClosePosition()` عند الفشل.

**فجوة حرجة - مراكز Agent مستثناة:**

منطق Break-Even و Trailing Stop يقعان بعد كتلة الإرجاع المبكر لمراكز Agent (الأسطر 352-393). هذا يعني أن مراكز Agent تحصل على فحوصات SL/TP الأساسية فقط، لكنها لا تحصل أبدًا على حماية Break-Even أو Trailing Stop. معلمات Trailing (تفعيل عند 2%، مسافة 1.2%) ثابتة ولا يمكن تهيئتها. كما أن الثابت `MONITOR_INTERVAL_MS` (10 ثوانٍ) معرَّف لكن غير مُستخدَم - الفاصل الفعلي هو 30 ثانية من الديكوريتور `@Interval` .

## #17: فلتر تكرار الصفقات

تم تنفيذ ثلاث قواعد عبر مفاتيح Redis في RiskGatekeeper:

| القاعدة | المفتاح | المنطق |
|---|---|---|
| قفل الاتجاه | trade-rep:dir-lock:{userId}:{symbol}:{side} | حظر نفس الاتجاه لمدة 30 دقيقة بعد الإغلاق |
| الحد اليومي | trade-rep:daily:{userId}:{symbol} | أقصى 5 صفقات لكل رمز يوميًا |
| الخسائر المتتالية | trade-rep:consec-loss:{userId}:{symbol} | 3 خسائر متتالية = حظر ساعتين |

يتم تحديث جميع المفاتيح ذريًا عند إغلاق المركز في `trading.service.ts` . الخسائر المتتالية تُصفَّر عند الربح. القواعد تعتمد كليًا على Redis - إذا تعطل Redis، يُسمح بالتداول (فشل مفتوح). Cooldown الخاص بـ Position Monitor (5 دقائق، جميع الاتجاهات) وقفل الاتجاه (30 دقيقة، نفس الاتجاه فقط) يخدمان أغراضًا متداخلة لكن بنطاقات مختلفة.

> **ملاحظة:**
> لا يوجد فلتر تكرار عبر الرموز. يمكن للمستخدم فتح 5 مراكز على 5 رموز مختلفة (كلها تحت الحد اليومي) خلال دقائق. كما أن جميع المعلمات (30 دقيقة، 5/يوم، 3 خسائر، حظر ساعتين) ثابتة غير قابلة للتهيئة.

# المرحلة الرابعة: التطوير الهيكلي

الأسبوع 5-8 | 4 تطويرات معمارية | 1/4 مكتمل

| # | التطوير | الحالة | التعقيد المتوقع | الأولوية |
|---|---|---|---|---|
| 18 | توحيد مساري V1 و V2 | غير مُنفَّذ | عالي (3-5 أيام) | عالية - دين هيكلي |
| 19 | تقسيم AIOrchestrator | غير مُنفَّذ | متوسط-عالي (3-4 أيام) | متوسطة - ملف 1667 سطر |
| 20 | توحيد المؤشرات المكررة | جزئي | منخفض-متوسط (1-2 يوم) | متوسطة - خطر صيانة |
| 21 | نظام رصد الأداء الفوري | مُنفَّذ | منخفض (فجوات فقط) | منخفضة - تحسينات |

## #18: توحيد مساري V1 و V2 - غير مُنفَّذ

يوجد حاليًا مساران مستقلان تمامًا للتداول مع بنى مختلفة جذريًا:

### V1: TradingController

**المسار:** `@Controller('trading')`

**التنفيذ:** متزامن - استدعاء مباشر لـ TradingService

**Idempotency:** شبه حتمي: `v1-{userId}-...-{symbol}-{side}`

**الطوابير:** لا توجد - تنفيذ مباشر

**المزايا الحصرية:** Force-close، تحديث مستويات المركز، سجل التداول، معلمات المخاطر

### V2: OrderController

**المسار:** `@Controller('trading/v2')`

**التنفيذ:** غير متزامن - آلة حالة + طابور

**Idempotency:** مفتاح صريح من العميل + IdempotencyService

**الطوابير:** BullMQ + RabbitMQ مع احتياطي

**المزايا الحصرية:** Idempotency صحيح، آلة حالة، ملخص محفظة

**مخاطر دين البيانات:**

المساران يكتبان إلى نفس قاعدة البيانات بتعيينات حقول مختلفة. هذا يخلق خطر تباعد البيانات. V1 يحتوي ميزات لا يملكها V2 (Force-close، تحديث المستويات) والعكس صحيح (Idempotency، آلة الحالة). لا يوجد كود مشترك بين المسارين. التوحيد يتطلب اختيار مسار واحد (يفضَّل V2 لبناه غير المتزامنة) وتحويل جميع نقاط النهاية إليه.

## #19: تقسيم AIOrchestrator - غير مُنفَّذ

AIOrchestratorService لا يزال ملفًا ضخمًا يتكون من 1,667 سطرًا يحتوي على جميع مسؤوليات التنسيق الذكي. يحقن 13 خدمة كتبعيات ويتحمل المسؤوليات التالية:

| المسؤولية | الأسطر | الخدمة المُقترَحة |
|---|---|---|
| توجيه النماذج + Circuit Breaker | ~170 | ModelRouterService |
| إجماع AI Council (8-10 أدوار) | ~600 | CouncilService (مجلد strategic-council/ موجود) |
| حارس ميزانية Bedrock | ~30 | BudgetGuardService |
| تحليل النموذج الواحد | ~200 | يبقى في AIOrchestrator (واجهة رقيقة) |
| تحليل الثقة الديناميكي | ~50 | يبقى في AIOrchestrator |
| أخرى (تسجيل، حالة، بيانات سوق) | ~617 | موزعة على الخدمات المستخرجة |

المجلد `/strategic-council` موجود مسبقًا مع StrategicCouncilService و StrategicCouncilModule و StrategicCouncilController، لكن منطق الإجماع لا يزال في AIOrchestrator بدلاً من التفويض إليه. هذا يعني أن البنية التحتية للاستخراج موجودة مسبقًا ويمكن البدء فورًا بنقل منطق الإجماع (أكبر كتلة بـ ~600 سطر).

## #20: توحيد المؤشرات المكررة - جزئي

يوجد تنفيذان مستقلان لكل مؤشر فني:

| المؤشر | indicator-algorithms.util.ts (المرجعي) | indicators.service.ts (الخدمة) | التطابق |
|---|---|---|---|
| SMA | calcSma() | this.sma() | متطابق |
| EMA | calcEma() / calcEmaLatest() | this.ema() | متطابق |
| RSI | calcRsi() - تنعيم Wilder | this.rsi() + تفسير | متطابق + إثراء |
| MACD | calcMacd() / calcMacdScalar() | this.macd() + نتيجة | متطابق + إثراء |
| Bollinger | calcBollingerBands() | this.bollingerBands() | متطابق + إثراء |
| ATR | calcAtr() / calcAtrLatest() | this.atr() + مستوى تقلب | متطابق + إثراء |

الخوارزميات متطابقة لكن المؤشرات الخدمية لا تستورد من المرجعية. ملف `indicator-algorithms.util.ts` يصرح صراحة: "التنفيذات المرجعية... لا تُدخِل نسخًا مبسطة هنا - هذا هو المصدر الوحيد للحقيقة." ومع ذلك، `TechnicalIndicatorService` لا يستورد منه - لديه نسخه الخاصة. المستوردون من المرجعي: market-data.service.ts (AI) و market-analyzer.service.ts (agents). المستوردون من الخدمي: analytical-ai.service.ts و signal-generator.service.ts و scanner.service.ts.

> **مخاطر الصيانة:**
> أي إصلاح يجب تطبيقه في مكانين. لا يوجد تباعد فعلي بعد (الخوارزميات متطابقة)، لكن العبء مضاعف. الحل المقترح: يجعل TechnicalIndicatorService يستدعي الدوال المرجعية ويضيف فقط الإثراء الخاص به (التفسير، التسجيل، الملخص). هذا تعديل مباشر يمكن إنجازه في 1-2 يوم.

## #21: نظام رصد الأداء الفوري - مُنفَّذ

يوجد نظامان مكملان يوفران رصد الأداء الفوري:

### PerformanceTrackerService (283 سطر)

مقاييس مجمعة: معدل الفوز، معامل الربح، نسبة Sharpe، أقصى تراجع، معيار Kelly (نصف Kelly)

تحديث مجدول كل 30 دقيقة عبر Cron مع تخزين مؤقت Redis بـ TTL 30 دقيقة

إيقاف تلقائي عند خسارة يومية 5% وتحديد المستخدمين النشطين

حساب حجم مركز Kelly للاستخدام من OrderDispatcher

### PerformanceEventsService (339 سطر)

تتبع فوري للأحداث: recordTradeClosed() يُستدعى عند كل إغلاق مركز

مجاميع جارية: PnL يومي، معدل فوز، إجمالي الصفقات محدثة في كل إغلاق

لقطات أداء مؤقتة في Redis (TTL 5 دقائق) كمصدر وحيد للحقيقة

سجل أحداث آخر 24 ساعة عبر getRecentTradeEvents()

**الفجوات المتبقية (غير حرجة):**

لا يوجد دفع WebSocket/SSE للواجهة الأمامية - لوحة القيادة تحتاج للاستقصاء. لا يوجد نظام تنبيه (بريد، إشعار) عند اقتراب الحدود. لا توجد نقطة نهاية Controller مخصصة للواجهة الأمامية. بيانات جزئية في `getRecentTradeEvents()` حيث بعض الحقول مثل `pnlPercent` و `holdingDurationMs` تأخذ قيمًا صفرية لأن جدول الصفقات لا يحتوي على بيانات مستوى المركز.

# الفجوات الحرجة والمخاطر

## فجوات عالية الخطورة

### 1. تحيز اتجاه البيع غير معالَج بالكامل (#14)

الآلية الحالية (تعزيز 1.3x في المجلس الاستراتيجي) لا تؤثر على مسار Agent الذي يولد 85%+ من صفقات البيع. خدمات SignalEvaluator و AdaptiveStrategySelector تعمل بدون أي وعي بالتوازن الاتجاهي. لا يوجد حد أقصى صارم لنسبة البيع. النتيجة: التحيز قد يستمر رغم التحسين الجزئي.

### 2. مسارات V1/V2 غير موحدة (#18)

مساران مختلفان هيكليًا يكتبان لنفس قاعدة البيانات. لا كود مشترك. V1 متزامن وV2 غير متزامن. كل مسار يملك ميزات حصرية. هذا دين هيكلي يزداد سوءًا مع كل إضافة جديدة ويزيد خطر تباعد البيانات.

### 3. مراكز Agent مستثناة من Break-Even و Trailing Stop (#16)

كتلة الإرجاع المبكر لمراكز Agent تمنع وصولها لمنطق Break-Even و Trailing Stop. هذه الميزات أساسية لحماية الأرباح وتقليل المخاطر، لكن Agent يعمل بدونها. هذا يعني أن مراكز Agent يمكن أن تتحول من رابحة إلى خاسرة دون أي حماية.

### 4. مفتاح قفل الاتجاه غير مكتوب (#11)

مفتاح `trade-rep:dir-lock` Redis `...:` يُفحَص لكن لا يُكتَب عند إغلاق المركز. القاعدة الأولى لفلتر التكرار (قفل 30 دقيقة) قد تكون غير فعالة عمليًا. القاعدتان الأخرتان (الحد اليومي والخسائر المتتالية) تعملان بشكل صحيح.

### 5. BullMQ Singleton Guard غير مكتمل (#4)

الحارس الحالي يسجل تحذيرًا فقط لكن المعالج المكرر لا يزال يعالج المهام. في بيئة hot-reload يمكن أن يحدث معالجة مزدوجة لأوامر التداول مما يؤدي لتنفيذ نفس الأمر مرتين.

## فجوات متوسطة الخطورة

### 6. تعارض عتبات R:R (#13)

Agent يستخدم عتبات استراتيجية (0.4-1.5) بينما RiskGatekeeper يفرض 1.5 كحد أدنى. العتبات الاستراتيجية تُلغَى فعليًا.

### 7. نطاقات PRICE_SANITY ثابتة (#9)

النطاقات السعرية ثابتة ولا تشمل جميع الرموز (~10 فقط). ستصبح قديمة مع تغير الأسعار.

### 8. خصم الهامش خارج المعاملة (#1)

الخصم يحدث بعد المعاملة وليس داخلها. تعطل بين المعاملة والخصم = مركز مفتوح بدون هامش.

### 9. AIOrchestrator ضخم (#19)

1667 سطر، 13 تبعية محقونة. صعوبة الصيانة والاختبار. منطق الإجماع (~600 سطر) يجب تفويضه لـ StrategicCouncilService.

## فجوات منخفضة الخطورة

### 10. مؤشرات مكررة (#20)

خوارزميات متطابقة في ملفين. خطر صيانة مضاعف لكن لا تباعد فعلي بعد.

### 11. لا دفع فوري للأداء (#21)

نظام الرصد يعمل لكن لا يوجد WebSocket/SSE للواجهة الأمامية. الاستقصاء مطلوب.

# الخلاصة والتوصيات

## ملخص التقييم

| المرحلة | التنفيذ | الجودة | المخاطر المتبقية |
|---|---|---|---|
| الأولى: الإصلاح الطارئ | 8/8 مكتمل | قوية مع 3 تنبيهات | BullMQ singleton، سباق الهامش، أوامر Limit عند انتهاء cache |
| الثانية: تصحيح البيانات | 4/4 مكتمل | جيدة مع تحفظات | نطاقات ثابتة، كاشف راكد Paper فقط، مفتاح قفل الاتجاه |
| الثالثة: التحسين الخوارزمي | 4/5 جزئي | متوسطة | تحيز البيع غير معالج بالكامل، مراكز Agent مستثناة من Trailing |
| الرابعة: التطوير الهيكلي | 1/4 مكتمل | منخفضة | دين AIOrchestrator، V1/V2 ضخم، مؤشرات مكررة |

## خريطة طريق الإصلاحات الموصى بها

### أولوية قصوى (خطر مالي مباشر)

| # | الإصلاح | الجهد | التأثير |
|---|---|---|---|
| A1 | إصلاح قفل اتجاه فلتر التكرار (كتابة المفتاح عند الإغلاق) | 1 ساعة | منع تكرار الصفقات المتكررة |
| A2 | تمديد Break-Even و Trailing Stop ليشمل مراكز Agent | 3 ساعات | حماية أرباح Agent من التراجع |
| A3 | إصلاح BullMQ Singleton Guard لمنع التسجيل المكرر فعليًا | 2 ساعة | منع المعالجة المزدوجة |
| A4 | نقل خصم الهامش داخل المعاملة | 2 ساعة | منع مراكز مفتوحة بدون هامش |

### أولوية عالية (تأثير على جودة التداول)

| # | الإصلاح | الجهد | التأثير |
|---|---|---|---|
| B1 | معالجة تحيز البيع في Agent (إضافة وعي بالتوازن الاتجاهي) | 1-2 يوم | توزيع أفضل للصفقات BUY/SELL |
| B2 | توحيد عتبات R:R بين RiskGatekeeper و Agent | 4 ساعات | اتساق في معايير المخاطر |
| B3 | إضافة نطاقات PRICE_SANITY ديناميكية أو تغطية أوسع | 4 ساعات | منع أسعار فاسدة لرموز إضافية |

### أولوية متوسطة (دين هيكلي)

| # | الإصلاح | الجهد | التأثير |
|---|---|---|---|
| C1 | توحيد مساري V1 و V2 | 3-5 أيام | إزالة دين البيانات والكود المكرر |
| C2 | تقسيم AIOrchestrator (استخراج CouncilService و ModelRouterService) | 3-4 أيام | قابلية الصيانة والاختبار |

| # | الإصلاح | الجهد | التأثير |
|---|---|---|---|
| C3 | توحيد المؤشرات المكررة | 1-2 يوم | مصدر واحد للحقيقة |

### الخلاصة النهائية

المرحلتان الأولى والثانية مُنجَزتان بجودة عالية مع بعض التحفظات المهمة. المرحلة الثالثة مُنجَزة جزئيًا مع فجوة كبيرة في معالجة تحيز اتجاه البيع. المرحلة الرابعة بأكملها تقريبًا لم تبدأ بعد باستثناء نظام رصد الأداء. يُوصَى بإكمال الإصلاحات ذات الأولوية القصوى فورًا (إجمالي ~8 ساعات عمل)، ثم الانتقال لمعالجة تحيز البيع وتوحيد العتبات قبل البدء في التطوير الهيكلي الكبير للمرحلة الرابعة. التأخير في معالجة الفجوات الحرجة (خصوصًا تحيز البيع واستثناء Agent من Trailing) يعرض النظام لمخاطر مالية حقيقية.