

AI 트레이딩 어시스턴트 V6.0: 최종 시스템 설계도

문서 버전: 6.0 (Final Blueprint)

작성일: 2025년 8월 15일

프로젝트 책임자: AI 어시스턴트 (Chief Strategist)

제1장: 통합 철학: 프로덕션 등급의 진화하는 트레이딩 인텔리전스

본 문서는 AI 트레이딩 어시스턴트 프로젝트의 모든 이전 기획 단계(V3.1, V4, V5.0, V5.1, V5.5)를 통합하고 최종 확정하는 단일 마스터 설계도(Single Source of Truth)이다. 프로젝트의 핵심 로직 데이터베이스(Atom_DB, Molecule_DB)는 최신화가 완료되었으며, 이제 우리의 과제는 분산된 아이디어들을 하나의 일관된 비전으로 종합하고, 개발팀이 즉시 실행에 옮길 수 있는 명확하고 구체적인 청사진을 제시하는 것이다.¹

이전 기획안의 복잡성에 대한 우려를 해소하기 위해, 본 설계도는 단순히 기능을 축소하는 것이 아니라 각 구성 요소의 존재 이유와 상호 연결성을 명확히 함으로써 그 복잡성을 정당화하고 논리적 필연성을 입증하고자 한다. 이 시스템의 최종 목표는 단순한 자동 매매 봇이 아니라, 인간의 전략적 통찰력과 기계의 무한한 처리 능력이 결합하여 함께 성장하고 진화하는, 프로덕션 등급의 지능형 트레이딩 파트너를 구축하는 것이다.

1.1. 핵심 비전: 헤드 셰프와 AI 수석 보조 셰프의 파트너십

우리가 구축하는 시스템의 본질은 V4 제안서에서 제시된 '헤드 셰프(Human Strategist)'와 'AI 수석 보조 셰프(AI Assistant)'의 파트너십 비전에 그 뿌리를 두고 있다.¹ 이 비유에서 인간 트레이더는 최종 의사결정권을 가진 '헤드 셰프'로서, 시장의 미묘한 분위기를 읽고 창의적인 통찰을 제공하며 AI의 성장을 감독하는 교사의 역할을 수행한다. AI 어시스턴트는 헤드 셰프의 지휘 하에 방대한 데이터를 학습하고, 패턴을 발견하며, 확률 높은 시나리오를 제안하고, 모든 경험을 기록하여 스스로를 개선하는 가장 유능한

파트너, 즉 '수석 보조 셰프'가 된다.¹

이러한 창의적이고 유기적인 파트너십이 성공적으로 작동하기 위해서는, 이들이 활동할 '주방'이 반드시 전문가 수준의 안정성과 신뢰성을 갖추어야 한다. 소프트웨어 공학에서 '프로덕션 레디(Production-Ready)' 시스템이란 단순히 기능이 구현된 상태를 넘어, 안정성, 견고성, 성능, 보안, 유지보수성 등 실제 운영 환경의 모든 요구사항을 만족하는 상태를 의미한다.² 불안정한 시스템은 유능한 보조 셰프가 될 수 없으며, 예측 불가능한 도구는 헤드 셰프의 창의성을 돕는 것이 아니라 방해할 뿐이다.

따라서 V5.5 기획안에서 제시된 '검증된 안전성과 지능형 자동화'라는 원칙과 그에 따른 WFO(Walk-Forward Optimization) 백테스팅, 검역-승인 워크플로우, 실시간 위험 관리 등의 프로덕션 등급 기능들은 V4의 인간 중심 비전을 대체하는 것이 아니라, 오히려 그것을 현실 세계에서 실현 가능하게 만드는 필수적인 기반 시설이다.¹ 예를 들어, '검역(Quarantine)' 시스템은 헤드 셰프가 새로운 '레시피(Molecule)'를 실험할 때 전체 '주방(포트폴리오)'을 위험에 빠뜨리지 않고 안전하게 테스트할 수 있도록 보장하는 '테스트 키친'과 같다. 이처럼 V6.0 시스템은 인간의 직관과 AI의 자동화가 프로덕션 등급의 견고한 프레임워크 안에서 시너지를 창출하는 것을 궁극적인 비전으로 삼는다.

1.2. 작동 원리: 상향식(Bottom-Up) 지식 발견 패러다임

시스템의 인지 과정은 V3.1 기획안에서 정립된 '상향식(Bottom-Up)' 패러다임을 따른다.¹ 이는 실제 트레이더가 시장을 관찰하고 가설을 세우는 인지 과정과 일치하며, 시스템의 학습 능력을 극대화한다.

1. **관찰 (Observe):** 시스템은 먼저 시장에서 발생하는 객관적이고 측정 가능한 최소 단위의 현상, 즉 '아톰(Atom)'을 관찰한다. '1분봉 20EMA 지지'나 '거래량 폭발'과 같은 아톰들은 그 자체로 매수/매도 신호가 아니라, 시장의 상태를 나타내는 순수한 사실(fact)에 해당한다.¹
2. **조합 (Combine):** 시스템은 관찰된 아톰들을 레고 블록처럼 조합하여 의미 있는 패턴, 즉 '분자(Molecule)'를 형성한다. 예를 들어, '시장 주도주'라는 컨텍스트 아톰과 '1분봉 EMA 응축'이라는 구조적 아톰, 그리고 'VWAP 골든크로스'라는 트리거 아톰이 특정 순서로 결합될 때 하나의 검증 가능한 트레이딩 가설, 즉 분자가 완성된다.¹
3. **해석 (Interpret):** 특정 분자가 왜 성공했거나 실패했는지에 대한 '이유(Why)'는 '컨텍스트(Context)', 즉 인간의 해석 영역에 속한다. AI가 '무엇(What)'이 일어났는지를 데이터로 보여주면, 헤드 셰프는 그 배경에 있는 서사(narrative)나 시장 심리를 해석하여 시스템의 지능을 한 차원 높인다.

이 상향식 접근법은 "A급 호재가 있으니 B패턴을 찾자"는 식의 하향식 접근법이 가졌던

확증 편향의 위험을 제거한다.¹ 대신, 데이터가 먼저 말하게 하고 인간은 그 의미를 해석함으로써, 시스템은 끊임없이 새로운 지식을 발견하고 스스로를 개선하는 진정한 학습 엔진으로 작동하게 된다.

제2장: 지식 아키텍처: 시장 현상에서 실행 가능한 분자까지

시스템의 모든 지능은 시장을 이해하고 반응하는 언어 체계, 즉 '지식 아키텍처'에 기반한다. 이 아키텍처는 시장의 모든 현상을 분해한 최소 단위인 '아톰'과, 이 아톰들을 조합하여 만든 실행 가능한 전략인 '분자'로 구성된다. 본 섹션에서는 최종 확정된 Atom_DB와 Molecule_DB를 기반으로 시스템의 핵심 지적 자산을 명확히 정의한다.¹

2.1. 아톰 렉시콘(Atom Lexicon): 시장의 알파벳

아톰은 더 이상 분해할 수 없는, 객관적이고 기계가 읽을 수 있는(machine-readable) 시장의 기본 사건이다.¹ 이는 시스템이 시장을 '읽는' 데 사용하는 알파벳과 같다. 아톰은 그 성격과 시간적 스케일에 따라 다음과 같은 네 가지 계층으로 분류되며, 이는 분석의 효율성과 정교함을 높인다.¹

- **컨텍스트 아톰 (Context, CTX):** 시장의 '날씨'와 같이 느리게 변하는 거시적 상태를 나타낸다. '나스닥 상승추세'(CTX-015)나 '시장 주도주'(CTX-010) 등이 여기에 해당하며, 특정 패턴의 성공 확률에 지대한 영향을 미친다.¹
- **구조적 아톰 (Structural, STR):** 차트의 '지형'에 해당하는 특정 패턴이나 지지/저항 레벨을 의미한다. '1분 EMA 응축'(STR-001)이나 '이전 고점 지지(S/R Flip)'(STR-015)처럼, 구체적인 움직임이 임박했음을 암시하는 '셋업(Setup)' 단계의 아톰들이다.¹
- **트리거 아톰 (Trigger, TRG):** 시장의 잠재적 변화를 알리는 '방아쇠' 역할을 하는 동적인 사건이다. '1분 VWAP 골든크로스'(TRG-001)나 '거래량 폭발'(TRG-003)과 같이, 즉각적인 행동을 유발할 수 있는 신호다.¹
- **파생 아톰 (Derived, DRV):** 여러 개의 기본 아톰이 특정 시공간에 합쳐져 만들어지는 강력한 융합 신호다. '1분 20EMA와 5분 100EMA의 컨버전스'(DRV-001)와 같이, 단일 지표가 아닌 여러 지지/저항 요소가 겹치는 '컨버전스 존'을 나타내며 매우 높은 신뢰도를 갖는다.¹

최종 확정된 아톰 렉시콘은 아래 표 2.1과 같다. 이 표는 개발팀이 시스템의 모든 감지

로직을 구현할 때 참조해야 하는 유일하고 명확한 정의서 역할을 한다.

표 2.1: 최종 아톰 렉시콘 (The Final Atom Lexicon)

Atom_ID	Atom_Name	Category	Description	Output_Column_Name
CTX-001	촉매_A++등급	Context	뉴스의 등급이 'A++'로 분류됨	is_catalyst_Aplusplus
CTX-005	일봉_하락추세	Context	일봉 차트가 명백한 하락 추세(주요 이평선 역배열)에 있음	is_daily_downtrend
CTX-010	시장_주도주	Context	당일 거래대금 및 상승률 최상위로 시장의 수급을 독점하는 종목	is_market_leader
CTX-014	시간외_과열	Context	프리마켓 또는 애프터마켓에서 주가가 과도하게 상승(예: 50% 이상)	is_aftermarket_overheat
STR-001	1분_EMA_응축	Structural	1분봉 5, 20, VWAP 등 단기 이평선들이 1% 이내로 수렴	is_1m_ema_compression

STR-003	1분_20EMA_지지	Structural	1분봉 캔들 저가가 1분봉 20EMA에 1% 이내 근접하며 지지	is_supported_at_1m_ema20
STR-007	1분_VWAP_지지	Structural	1분봉 캔들 저가가 VWAP 위에서 1% 이내 근접하며 지지	is_supported_at_1m_vwap
STR-015	이전_고점_지지	Structural	이전의 의미있는 고점(저항선)이 현재 지지선으로 작용(S/R Flip)	is_supported_at_prev_high
TRG-001	1분_VWAP_골든크로스	Trigger	1분봉 20EMA가 VWAP을 상향 돌파 (최근 5분 이내)	triggered_1m_vwap_gc
TRG-003	거래량_폭발	Trigger	일정 기간(예: 5분) 평균 대비 상대 거래량이 500% 이상 급증	triggered_volume_spike
TRG-007	VWAP_붕괴	Trigger	주가가 1분봉 VWAP 지지선을 하향 이탈	triggered_vwap_breakdown
DRV-001	컨버전스_1m20_5m100	Derived	1분 20EMA와 5분 100EMA가 1% 가격 범위	is_conv_1m20_5m100

			내에서 겹침		
DRV-003	컨버전스_SR 플립_1m50	Derived	이전 고점(저항)과 1분 50EMA가 1% 가격 범위 내에서 겹침	is_conv_srfl p_1m50	
(주: 공간 제약으로 일부 아톰만 표시. 전체 목록은 Atom_DB.csv 1 참조)					

2.2. 분자 컴펜디움(Molecule Compendium): 전략 레시피 북

분자는 아톰들을 조합하여 만든 하나의 완성된 트레이딩 전략, 즉 '전략 레시피'이다.¹

Molecule_DB는 이러한 분자들의 집합체로서, 시스템의 실제적인 행동 지침을 담고 있는 전략 사전(Playbook) 역할을 한다.

분자 정의의 핵심 요소 중 하나는 Translation_Notes 컬럼이다. Required_Atom_IDs가 전략의 '뼈대(what)'를 정량적으로 정의한다면, Translation_Notes는 그 전략의 '영혼(why and how)'에 해당하는 질적 뉘앙스와 인간의 통찰을 담는다.¹ 예를 들어,

LOGIC-AVD-004(불 트랩) 분자는 단순히 '1분 정배열 형성'(TRG-008)이라는 아톰을 찾는 것이 아니다. Translation_Notes는 이 신호가 '일봉 하락추세'(CTX-005)와 '약한 거래량 돌파'(STR-013)라는 부정적 맥락과 충돌할 때, 이는 신뢰도 낮은 '함정'임을 명시적으로 AI에게 알려준다.¹ 이처럼

Translation_Notes는 인간 헤드 셰프의 지혜를 AI 보조 셰프에게 전달하는 핵심적인 통로이다.

최종 확정된 분자 컴펜디움의 일부는 아래 표 2.2와 같다.

표 2.2: 최종 분자 컴펜디움 (The Final Molecule Compendium)

Molecule_ID	Molecule_Name	Category	Required_At om_IDs	Translation_ Notes
LOGIC-AVD-001	시간 외 과열 후 본장 개장 급락	회피/위험관 리	CTX-005, CTX-014, TRG-002	'떨어지는 칼날' 패턴. 시간 외 과열로 인한 차익 실현 매물과 장기 추세 매도세가 결합된 강력한 매도 압력. 모든 매수 신호를 무시해야 함. (TELO 사례 참고)
LOGIC-AVD-004	악한 거래량의 정배열 돌파 (불 트랩)	회피/위험관 리	CTX-005, STR-013, TRG-008	'불 트랩' 패턴. 거시적 하락 추세 속에서 악한 거래량으로 형성된 미시적 상승 신호는 신뢰도 매우 낮음. (TMDE 06/23 사례 참고)
LOGIC-AVD-010	VWAP 붕괴 후 성급한 반등 매수	회피/위험관 리	CTX-010, TRG-007	주도주가 힘을 잃고 VWAP을 하향 이탈했을 때, 첫 반등 시도는 VWAP 저항에 막혀 실패할 확률이 매우 높음. (LIDR 오답노트 기반 생성)

LOGIC-EXP-002	VWAP 골든크로스 후 컨버전스 존 놀림목	반등/진입	CTX-003, CTX-015, DRV-001, TRG-001	1차 상승 파동 이후, 1분봉과 5분봉의 핵심 이평선이 겹치는 지지 구간. 신뢰도 매우 높은 놀림목. (AVBE 사례 기반)	
LOGIC-EXP-018	이전 고점(S/R 플립) + 1분봉 50EMA 지지	반등/진입	CTX-010, DRV-003	며칠 전의 의미있는 고점이 지지선으로 작용하는 S/R 플립과 1분 50EMA가 겹치는 강력한 놀림목. (YHC 07/16 사례 기반)	
LOGIC-EXP-024	A++급 호재 당일 1분봉 VWAP 지지	반등/진입	CTX-001, CTX-010, STR-007	A++급 호재는 일반적인 개장 급락(AVD-001)을 무력화시키고 VWAP을 강력한 지지선으로 삼는 경향이 있음. AVD-001에 대한 예외 규칙. (ABVX 오답노트 기반 생성)	
(주: 공간 제약으로 일부 분자만 표시. 전체					

3.1. Layer 1 & 2: 데이터 수집 및 신호 생성 (SignalScanner 엔진)

- 기능: 이 두 계층은 시스템의 '눈과 귀' 역할을 하며, 시장 데이터를 실시간으로 수집하고 이를 시스템이 이해할 수 있는 최소 단위의 정보, 즉 '아톰'으로 변환한다. 이는 V5.0에서 제안된 SignalScanner 엔진의 핵심 기능을 공식화하고 확장한 것이다.¹
- 구현 명세:
 - 하이브리드 **API 전략: V5.1** 기획안의 해결책을 채택하여, 데이터의 성격에 따라 최적의 API를 사용한다.¹
 - 실시간 데이터: 실시간 예측의 핵심인 SignalScanner는 **Alpaca API**의 **WebSocket** 스트림을 사용한다. 이를 통해 1분봉, 5분봉 등 다수의 시간대 데이터를 동시에 구독하여 지연 없는 다중 시간대 분석을 수행한다.⁵
 - 과거 데이터: 비실시간 분석 및 학습을 수행하는 LogicDiscoverer와 MetaLearner는 방대한 과거 데이터 접근을 위해 **yfinance API**를 활용한다.¹
 - 아톰 탐지 및 기록: AtomDetector 컴포넌트는 실시간 데이터 스트림을 지속적으로 스캔하며, 표 2.1의 Atom_DB에 정의된 모든 아톰의 발생 여부를 검사한다. 아톰이 탐지되는 즉시, 해당 이벤트는 타임스탬프, 가격, 거래량 등 상세 정보와 함께 ****SIDB (Signal Instance Database)****에 기록된다. 이는 V3.1에서 개념화된 '잃어버린 연결고리'를 구현하는 것이며¹, V4에서 요구된 '모든 활동 기록' mandate를 완벽하게 충족시킨다.¹

3.2. Layer 3: 검증 시스템 (The Validation Gauntlet)

- 기능: 이 계층은 시스템의 '품질 관리 및 안전 심의 위원회' 역할을 한다. 인간이나 AI가 제안한 어떠한 새로운 전략('분자')도 이 계층의 엄격한 검증 절차를 통과하지 않고는 절대로 실 운영 환경에 배포될 수 없다. 이는 시스템의 안정성을 보장하는 핵심적인 안전장치이다.
- 구현 명세: 이 계층은 V5.5에서 제안된 프로덕션-레디 컴포넌트들로 구성된다.¹
 - **QuarantineManager:** LogicDiscoverer에 의해 생성된 모든 신규 분자는 status: 'quarantined' 상태로 Molecule_DB에 추가되고, 즉시 Quarantine_Queue 데이터베이스로 보내져 관리된다.¹
 - **WFOValidator:** Quarantine_Queue에 대기 중인 분자들을 대상으로 **Walk-Forward Optimization (WFO)** 백테스팅을 수행한다. WFO는 과거 데이터를 여러 구간으로 나누어 순차적으로 최적화와 검증을 반복함으로써, 단순 백테스트가 야기하는 과최적화(overfitting) 문제를 방지하고 전략의 강건성(robustness)을 평가하는 고급 검증 기법이다.¹¹ 모든 상세한 검증 결과는

WFO_Results 데이터베이스에 기록된다.¹

- **ApprovalWorkflow:** 인간 '헤드 셰프'가 WFO 검증 결과를 검토하고, 해당 분자를 실 운영에 투입할지 여부를 최종 결정하는 인간-개입(human-in-the-loop) 워크플로우이다. 사용자가 '승인(approve)'하면 해당 분자의 status는 active로 변경되며, '거부(reject)'하면 deprecated 상태가 된다. 모든 결정 행위와 사유는 Approval_Log에 영구적으로 기록되어 완전한 감사 추적성을 보장한다.¹

3.3. Layer 4: 실행 관리 (The Operations Core)

- **기능:** 이 계층은 검증을 통과한 '활성(active)' 상태의 분자들이 실제 시장에서 작동하는 라이브 운영 환경을 관리한다. 환경 분리, 실시간 매칭, 자동 위험 관리를 책임진다.
- **구현 명세:**
 - **EnvironmentManager:** 실제 자본이 투입되는 Production 환경과 안전한 테스트를 위한 Staging 환경을 완벽하게 분리한다. 각 환경은 별도의 데이터베이스 시트 ID, API 키, 리스크 한도 설정을 가지므로, 개발 및 테스트 과정이 운영 환경에 영향을 미치는 것을 원천적으로 차단한다.¹
 - **MoleculeMatcher:** 시스템의 핵심 실행 엔진. Layer 2에서 실시간으로 생성되는 SIDB 로그 스트림을 감시하며, Molecule_DB에 status: 'active'로 등록된 분자들의 Required_Atom_IDs 조합과 일치하는 패턴이 발생하는지를 실시간으로 매칭한다. 매칭이 발생하면 Prediction_Notes에 예측 기록을 남기고 사용자에게 알림을 보낸다.
 - **RiskMonitor:** 자동화된 '감사관' 역할을 수행한다. Performance_Dashboard의 실시간 성과 지표(예: 최대 손실폭, 승률)를 지속적으로 모니터링한다.¹ 만약 특정 분자의 성과가 미리 설정된 위험 임계치(예: Max Drawdown > 20%)를 하회하면, RiskMonitor는 해당 분자의 status를 자동으로 deprecated로 변경하여 추가적인 신호 생성을 즉시 중단시킨다. 이 모든 자동 개입 조치는 Risk_Alerts 데이터베이스에 상세히 기록된다.¹

3.4. Layer 5: 학습 및 진화 (LogicDiscoverer & MetaLearner 엔진)

- **기능:** 이 계층은 시스템의 '두뇌'이자 '헤드 셰프'와의 주된 소통 창구이다. 새로운 전략을 발견(LogicDiscoverer)하고 기존 전략을 복기하며 개선(MetaLearner)하는,

시스템의 지능이 성장하고 진화하는 공간이다.

- 구현 명세: V5.0에서 제안된 인간-AI 협력 엔진들이 이 계층에서 공식적으로 구현된다.¹
 - **LogicDiscoverer:** 사용자가 과거 특정 종목/날짜의 차트를 보며 얻은 통찰(예: "이평선 응축 후 돌파가 핵심이었음")을 자연어로 입력하면, 시스템은 해당 시점의 SIDB 로그와 가격 데이터를 Gemini API로 전송하여 분석을 요청한다. Gemini는 이 정보를 바탕으로 구조화된 신규 '분자'(Required_Atom_IDs와 Translation_Notes의 조합)를 제안한다. 사용자가 이를 승인하면, 이 신규 분자는 status: 'quarantined' 상태로 Layer 3의 검증 시스템으로 보내진다.
 - **MetaLearner:** 사용자가 Prediction_Notes에 기록된 과거 예측의 실제 결과('성공' 또는 '실패')를 입력하면, 시스템은 예측의 근거가 되었던 아톰들과 실제 시장에서 발생했던 아톰들의 전체 로그(SIDB 데이터)를 비교 분석한다. 이 '격차 분석(Gap Analysis)' 결과를 Gemini API를 통해 심층 분석하여 실패의 근본 원인을 진단하는 'AI 복기 리포트'를 생성한다. 이 리포트를 바탕으로 사용자는 기존 분자의 Translation_Notes를 수정하거나, 새로운 회피(AVD) 분자를 제안하는 등 시스템의 지식을 정교화할 수 있다.
 - **VersionController:** MetaLearner를 통해 발생하는 모든 분자 정의의 변경 사항은 VersionController에 의해 관리된다. 변경 전/후의 값, 변경자, 변경 사유 등 모든 이력이 Version_History 데이터베이스에 자동으로 기록되어, 모든 지식의 진화 과정을 완벽하게 추적하고 필요시 롤백할 수 있는 기반을 마련한다.¹

이 5계층 아키텍처는 단순한 기능의 나열이 아니라, 하나의 완결된 '지식 공장(Knowledge Factory)'을 형성한다. 인간의 직관적 통찰이 Layer 5에서 시스템에 주입되면, Layer 3의 엄격한 검증을 거쳐 Layer 4에서 실제 운영을 통해 데이터로 시험된다. 그 결과는 다시 Layer 1/2를 통해 기록되고, Layer 5에서 재분석되어 시스템을 더욱 정교하게 만드는 피드백 루프를 완성한다. 이것이 바로 우리가 진정으로 나아가야 할 방향, 즉 '지속적으로 학습하고 성장하는 트레이딩 인텔리전스'의 구체적인 설계도이다.

제4장: 데이터 인프라: 시스템의 중앙 신경계

사용자가 제기한 '구글 시트 부분의 간략화' 요청은 데이터베이스의 구성 요소들을 무조건적으로 제거하는 것이 아니라, 그들의 존재 이유와 상호 연결성을 명확히 하여 복잡성에 대한 논리적 타당성을 확보하는 것으로 해결해야 한다. V6.0의 데이터 인프라는 총 10개의 구글 시트(테이블)로 구성되며, 이는 3장에서 설명한 5계층 프로덕션 아키텍처의 각 기능 요소를 지원하기 위해 필수불가결하게 요구되는 최소한의 집합이다. 각 테이블은 독립적인 사일로가 아니라, 시스템 전체에 혈액과 신호를 공급하는 중앙

신경계처럼 유기적으로 연결되어 있다.

본 장에서는 10개의 데이터베이스 테이블을 5개의 논리적 그룹으로 재구성하여 그 구조와 목적을 명확히 제시한다.

4.1. 데이터베이스 스키마의 논리적 그룹화

1. **그룹 1: 핵심 로직 (The Brain):** 시스템의 근본적인 지식과 전략을 담는 저장소.
 - **Atom_DB:** 시장 현상의 최소 단위인 '아톰'을 정의하는 시스템의 사전.¹
 - **Molecule_DB:** 아톰들을 조합한 실행 가능한 트레이딩 전략 '분자'를 정의하는 시스템의 전략집.¹
2. **그룹 2: 운영 로그 (The Senses & Memory):** 시스템이 실시간으로 무엇을 보고, 듣고, 예측하는지를 기록하는 원시 데이터 로그.
 - **SIDB:** 시스템이 감지한 모든 아톰 발생 이벤트를 타임스탬프와 함께 기록하는 고충실도 로그. Layer 5의 학습 엔진(LogicDiscoverer, MetaLearner)이 과거를 분석하기 위한 가장 중요한 원재료이다.¹
 - **Prediction_Notes:** MoleculeMatcher가 특정 분자 로직에 기반하여 공식적인 '예측' 신호를 생성할 때마다 기록되는 로그. MetaLearner의 복기 및 성과 분석의 직접적인 입력 데이터가 된다.¹
3. **그룹 3: 검증 및 거버넌스 (The Quality Control Department):** 새로운 전략이 생성되어 실전에 투입되기까지의 생명주기를 관리하고 통제하는 테이블. Layer 3의 기능을 직접적으로 지원한다.
 - **WFO_Results:** WFOValidator가 수행한 모든 Walk-Forward Optimization 테스트의 상세 결과를 저장한다.¹
 - **Quarantine_Queue:** 검증 대기 중인 quarantined 상태의 분자 목록을 관리한다. WFOValidator의 작업 큐 역할을 한다.¹
 - **Approval_Log:** '헤드 셰프'가 검역 중인 분자를 승인 또는 거부한 모든 기록을 추적하는 불변의 감사 로그.¹
4. **그룹 4: 성과 및 리스크 (The Live Dashboard):** 실 운영 환경에서 전략들의 성과를 실시간으로 추적하고 위험을 관리하는 테이블. Layer 4의 기능을 직접적으로 지원한다.
 - **Performance_Dashboard:** 각 활성 분자의 핵심 성과 지표(KPI) - 승률, 손익비, 최대 손실폭 등을 실시간으로 집계하는 대시보드. RiskMonitor의 판단 근거가 된다.¹
 - **Risk_Alerts:** RiskMonitor가 성과 저하 분자를 자동으로 비활성화하는 등 위험 관리 조치를 발동했을 때 그 기록을 남기는 불변의 로그.¹
5. **그룹 5: 감사 및 버전 관리 (The Library of Record):** 시스템의 지적 자산에 대한

모든 변경 이력을 추적하여 완전한 투명성과 재현성을 보장하는 테이블.

- **Version_History:** Molecule_DB의 내용이 수정될 때마다 변경 전/후 값, 변경자, 시간, 사유를 기록하는 로그. **Google Apps Script**의 **onEdit** 트리거를 통해 자동화된다.¹

4.2. V6.0 통합 데이터 인프라 스키마

아래 표 **4.1**은 위에서 설명한 **10**개 테이블의 전체 스키마를 요약한 것이다. 이 표는 각 테이블이 어떤 아키텍처 계층을 지원하며, 어떤 목적을 가지고 있는지 명확하게 보여줌으로써, 데이터베이스의 복잡성이 시스템의 견고성과 지능을 위해 반드시 필요한 구조적 장치임을 증명한다.

표 **4.1: V6.0** 통합 데이터 인프라 스키마 (**The V6.0 Complete Data Infrastructure Schema**)

테이블 명	논리 그룹	아키텍처 지원 기능 (Layer)	컬럼 (예시)	컬럼 설명 (예시)
Atom_DB	핵심 로직	L2: AtomDetector	Atom_ID, Atom_Name, Description	시장 현상의 최소 단위 정의 ¹
Molecule_DB	핵심 로직	L3, L4, L5	Molecule_ID, Required_Atom_ IDs, Status	실행 가능한 전략 및 상태(quarantine d/active/deprec ated) 관리 ¹
SIDB	운영 로그	L2 -> L4, L5	Instance_ID, Timestamp_UTC , Atom_ID	모든 아톰 발생 이벤트를 기록하는 원시 데이터 ¹
Prediction_Notes	운영 로그	L4 -> L5	Prediction_ID, Triggered_Molec ule_ID, Actual_Outcome	분자 기반 예측 및 실제 결과 기록 ¹

Quarantine_Queue	검증/거버넌스	L5 -> L3	Queue_ID, Molecule_ID, WFO_Status	신규 생성된 분자의 검증 대기열 ¹
WFO_Results	검증/거버넌스	L3: WFOValidator	Result_ID, Molecule_ID, WFO_Efficiency	WFO 백테스팅 상세 결과 저장 ¹
Approval_Log	검증/거버넌스	L3: ApprovalWorkflow	Log_ID, Molecule_ID, Action, Reviewer	분자 승인/거부 결정에 대한 감사 로그 ¹
Performance_Dashboard	성과/리스크	L4: RiskMonitor	Molecule_ID, Win_Rate_%, Max_Drawdown_%	활성 분자의 실시간 성과 지표 집계 ¹
Risk_Alerts	성과/리스크	L4: RiskMonitor	Alert_ID, Molecule_ID, Alert_Type, Auto_Action	자동 위험 관리 조치 발동 기록 ¹
Version_History	감사/버전 관리	L5: VersionController	History_ID, Molecule_ID, Changed_Fields, Old_Values	Molecule_DB 변경 이력 완전 추적 ¹

이처럼 10개의 테이블은 각기 명확한 목적을 가지고 5계층 아키텍처의 특정 컴포넌트와 직접적으로 상호작용한다. SIDB가 없다면 MetaLearner는 과거를 복기할 수 없고, Performance_Dashboard가 없다면 RiskMonitor는 작동할 수 없으며, Quarantine_Queue가 없다면 '검증'이라는 개념 자체가 성립하지 않는다. 따라서 이 데이터 인프라는 현재 시스템의 비전을 달성하기 위한 최적의, 그리고 필수적인 구조이다.

제5장: 인간-AI 공생: 전략가의 워크플로우

시스템의 정교한 아키텍처와 데이터 구조는 그 자체로 목적이 아니라, 궁극적으로 '헤드 셰프', 즉 인간 전략가의 의사결정을 돕고 지능을 증강시키기 위한 도구이다. 따라서 V4와 V5.0 기획안에서 강조된 인간 중심의 설계 철학을 최종 설계도의 전면에 내세워야 한다.¹

본 장에서는 전략가가 일상적으로 시스템과 상호작용하며 새로운 지식을 창출하고 기존 지식을 개선하는 두 가지 핵심 워크플로우를 구체적으로 기술한다. 이는 시스템이 단순한 백엔드 프로세스가 아니라, 실용적이고 가치 있는 사용자 도구로 어떻게 구현되어야 하는지를 명확히 보여준다.

5.1. 워크플로우 1: 새로운 '레시피' 발견 (LogicDiscoverer 활용)

이 워크플로우는 인간의 창의적인 통찰력이 시스템의 새로운 지식으로 변환되는 과정이다.

- **1단계 (통찰):** 전략가는 자신의 주 거래 플랫폼에서 차트를 분석하던 중, 특정 종목의 성공 또는 실패 요인에 대한 '아하!' 순간을 경험한다. 예를 들어, "PROK 종목이 2025년 7월 9일에 성공했던 이유는, 장시간의 이평선 응축 이후 거래량을 동반하여 돌파했고, 이후 첫 20EMA 지지에서 반등했기 때문이다"라는 구체적인 가설을 세운다.¹
- **2단계 (상호작용):** 전략가는 시스템의 LogicDiscoverer UI를 연다. '티커(PROK)', '날짜(2025-07-09)', 그리고 1단계에서 얻은 '핵심 통찰'을 자연어 텍스트로 입력한 후 '분석 시작' 버튼을 클릭한다.¹
- **3단계 (AI 지원):** 백엔드는 요청을 받아, 해당 티커와 날짜의 과거 가격 데이터(yfinance 활용)와 당시 기록된 모든 아톰 로그(SIDB 데이터)를 검색한다. 이 정량적 데이터와 사용자가 입력한 정성적 텍스트를 하나의 패키지로 묶어 Gemini API에 전달하며, "이 정보를 바탕으로 새로운 '분자'를 구조화된 JSON 형식으로 제안하라"고 요청한다.
- **4단계 (제안):** Gemini API는 제공된 정보를 분석하여, 인간의 통찰을 아톰 언어로 번역한 결과를 반환한다. 예를 들어, 다음과 같은 JSON 형식의 제안을 생성할 수 있다:
 - Required_Atom_IDs: ``
 - Translation_Notes: "장시간의 에너지 응축(STR-001)과 거래량 폭발(TRG-003)을 동반한 돌파(TRG-004) 이후, 첫 번째 1분 20EMA 지지(STR-003)는 매우 신뢰도 높은 반등 타점이다."
- **5단계 (확정 및 검역):** UI는 이 제안을 사용자에게 명확하게 시각화하여 보여준다. 전략가는 제안된 내용을 검토하고, 필요시 수정 후 최종 '승인' 버튼을 클릭한다. 이 승인과 동시에, 새로운 분자는 Molecule_DB에 status: 'quarantined' 상태로 추가되고, 시스템의 Layer 3 '검증 시스템'으로 자동 이관되어 WFO 백테스팅 절차를 밟게 된다.

5.2. 워크플로우 2: 기존 '레시피' 개선 (MetaLearner 활용)

이 워크플로우는 실제 거래 결과를 바탕으로 시스템의 기존 지식을 복기하고 정교화하는 피드백 루프이다.

- **1단계 (복기 시작):** 전략가는 MetaLearner UI를 연다. UI에는 Prediction_Notes 테이블에서 가져온 최근 예측 목록이 표시된다.¹
- **2단계 (결과 입력):** 전략가는 복기할 예측 항목(예: "LIDR, LOGIC-EXP-004 신호 발생")을 선택하고, Actual_Outcome 필드에 실제 결과("실패 - VWAP 붕괴로 손절")를 입력하며, 추가적인 질적 피드백(Human_Feedback)을 남긴다.¹
- **3단계 (AI 분석):** '복기 시작' 버튼을 클릭하면, 백엔드는 해당 예측의 전체 컨텍스트를 재구성한다. 즉, 예측의 근거가 되었던 아톰들(Key_Atoms_Found), 예측 당시 누락되었던 필수 아톰, 그리고 예측 전후의 SIDB에 기록된 모든 주변 아톰들의 흐름을 종합한다. 이 방대한 컨텍스트 데이터와 사용자가 입력한 실제 결과를 Gemini API에 전달하여, "예측과 실제 결과 사이의 간극(Gap)에 대한 근본 원인을 분석하라"고 요청한다.
- **4. 단계 (실행 가능한 진단):** AI는 심층 분석 후 'AI Review Summary'를 생성한다. 예를 들어, "LOGIC-EXP-004의 조건은 모두 충족되었으나, 예측 직후 'VWAP 붕괴'(TRG-007) 아톰이 발생했다. 이는 '주도주가 힘을 잃었을 때의 첫 반등 시도는 위험하다'는 LOGIC-AVD-010 분자의 활성화 조건과 일치한다. 따라서 실패의 근본 원인은 더 높은 우선순위의 회피 신호를 간과했기 때문이다" 와 같은 구체적이고 실행 가능한 진단을 제공한다.
- **5단계 (시스템 진화):** 이 진단 리포트를 바탕으로, 전략가는 다음과 같은 조치를 취할 수 있다:
 - LOGIC-EXP-004 분자의 Translation_Notes에 "단, LOGIC-AVD-010이 활성화될 경우 이 신호는 무효화된다"는 예외 규칙을 추가한다.
 - LOGIC-AVD-010의 우선순위를 시스템적으로 더 높게 조정하도록 제안한다. 이러한 모든 변경 사항은 VersionController에 의해 Version_History에 기록된다. 또한, 이 거래의 실패 결과는 Performance_Dashboard에 즉시 반영되어 LOGIC-EXP-004의 실시간 성과 지표(승률, 수익률 등)를 업데이트한다. 이 과정을 통해 시스템은 실패로부터 배우고 스스로를 강화한다.

제6장: 구현 블루프린트: 프로덕션을 향한 단계별 로드맵

본 장에서는 AI 트레이딩 어시스턴트 V6.0을 아이디어에서 실제 작동하는 프로덕션 시스템으로 전환하기 위한 구체적이고 실행 가능한 개발 로드맵을 제시한다. 이 로드맵은 개발팀이 명확한 우선순위에 따라 체계적으로 프로젝트를 진행할 수 있도록 3단계의 페이지로 구성된다.

6.1. 단계별 구현 계획

Phase 1: 기반 구축 (Foundational Bedrock)

- **주요 목표:** 시스템의 핵심 데이터 구조를 확립하고, 새로운 전략을 안전하게 테스트하고 검증할 수 있는 핵심 엔진을 구축한다. 이 단계는 시스템의 뼈대를 만드는 과정이다.
- **핵심 과업:**
 1. **데이터 인프라 구축:** 제4장의 표 4.1에 명시된 10개의 구글 시트 스키마를 모두 생성하고, API 접근을 위한 인증 설정을 완료한다.
 2. **과거 데이터 이관 및 SIDB 채우기:** 기존의 거래 로그와 시장 데이터를 분석하여, 과거 데이터에 대한 AtomDetector를 실행하고 SIDB를 과거의 아톰 발생 기록으로 채운다.
 3. **WFOValidator 구현:** V5.5 기획안의 validator.py를 기반으로 Walk-Forward Optimization 백테스팅 엔진을 개발한다.¹ quarantined 상태의 분자를 입력받아 WFO 테스트를 실행하고, 그 결과를 WFO_Results 시트에 기록하는 기능을 구현한다.
 4. **검증 워크플로우 백엔드 구현:** QuarantineManager를 구현하여 Quarantine_Queue를 관리하고, ApprovalWorkflow의 백엔드 로직(분자 상태 변경, Approval_Log 기록 등)을 개발한다.

Phase 2: 시스템 강화 (Fortification)

- **주요 목표:** 시스템을 실시간 데이터에 연결하고, 자동화된 위험 관리 기능을 추가하며, 모든 변경 사항을 추적할 수 있는 견고한 운영 환경을 구축한다. 이 단계는

시스템에 갑옷과 무기를 장착하는 과정이다.

- 핵심 과업:

1. 실시간 **SignalScanner** 구현: Alpaca API의 WebSocket을 활용하여 실시간 시장 데이터를 수신하고, 이를 바탕으로 AtomDetector를 실행하여 SIDB에 실시간으로 로그를 기록하는 기능을 완성한다.
2. **EnvironmentManager** 구현: V5.5 기획안의 environment_manager.py를 기반으로 Staging과 Production 환경을 분리하고, 각 환경에 맞는 설정(API 키, DB ID 등)을 동적으로 로드하는 메커니즘을 구현한다.¹
3. **RiskMonitor** 구현: V5.5 기획안의 risk_monitor.py를 기반으로, Performance_Dashboard를 주기적으로 폴링하여 성과 저하 분자를 감지하고, 자동으로 deprecated 상태로 전환하며 Risk_Alerts에 기록하는 자동 위험 관리 모듈을 개발한다.¹
4. **VersionController** 및 감사 로그 구현: Molecule_DB의 변경 사항을 추적하는 VersionController(version_controller.py)를 개발하고, Google Apps Script의 onEdit 트리거를 설정하여 Version_History 시트에 변경 이력이 자동으로 기록되도록 구현한다.¹

Phase 3: 지능 부여 (Sentience)

- 주요 목표: 인간 전략가가 시스템과 직접 상호작용하며 시스템의 지능을 훈련시키고 성장시킬 수 있는 사용자 인터페이스와 AI 연동 기능을 구현한다. 이 단계는 시스템에 두뇌와 영혼을 불어넣는 과정이다.
- 핵심 과업:
 1. **LogicDiscoverer UI/UX** 및 **API** 개발: 제5.1장의 워크플로우를 지원하는 프론트엔드 UI(티커, 날짜, 텍스트 입력)와 백엔드 API(Gemini 연동, 분자 제안, 검역 큐 전송)를 개발한다.
 2. **MetaLearner UI/UX** 및 **API** 개발: 제5.2장의 워크플로우를 지원하는 프론트엔드 UI(예측 목록, 결과 입력)와 백엔드 API(Gemini 연동, 복기 리포트 생성, 성과 대시보드 업데이트)를 개발한다.
 3. 통합 대시보드 **UI** 개발: Performance_Dashboard의 데이터를 시각화하고, ApprovalWorkflow(검역 목록 확인 및 승인/거부)를 처리할 수 있는 통합 관리자 대시보드 UI를 구축한다.
 4. 전체 시스템 통합 테스트: 3개의 페이지에서 개발된 모든 컴포넌트를 통합하여, '통찰 -> 제안 -> 검역 -> 승인 -> 운영 -> 성과 저하로 자동 비활성화 -> 복기 -> 개선'에 이르는 전체 생명주기(Lifecycle)가 원활하게 작동하는지 엔드투엔드(End-to-End) 테스트를 수행한다.

6.2. 최종 구현 로드맵 요약

아래 표 6.1은 V6.0 시스템 구축을 위한 전체 개발 로드맵을 요약한 것이다. 이 표는 각 단계별 목표와 결과물을 명확히 하여 프로젝트 관리의 기준점을 제공한다.

표 6.1: V6.0 구현 로드맵 (The V6.0 Implementation Roadmap)

단계	주요 목표	핵심 개발 컴포넌트 / 파일	완료의 정의 (Definition of Done)
Phase 1	검증 시스템 기반 구축	- 10개 DB 스키마 생성 - AtomDetector (과거 데이터용) - WFOValidator (validator.py) - QuarantineManager - ApprovalWorkflow (백엔드)	- 신규 분자를 시스템에 제안하고, WFO 백테스팅을 거쳐, 인간의 승인을 통해 active 상태로 변경하는 전체 백엔드 파이프라인이 완성됨.
Phase 2	고급 검증 및 보안 강화	- SignalScanner (실시간 WebSocket) - EnvironmentManager - RiskMonitor (risk_monitor.py) - VersionController (version_controller.py) - Google Apps Script (onEdit)	- 시스템이 Staging/Production 환경에서 실시간으로 작동하며, 성과가 나쁜 전략을 자동으로 비활성화하고, 모든 로직 변경 이력을 추적할 수 있게 됨.
Phase 3	지능형 자동화 및 최적화	- LogicDiscoverer (UI + API) - MetaLearner (UI + API) - 통합 관리자 대시보드 (UI) - Gemini API 연동	- 사용자가 직관적인 UI를 통해 AI와 협력하여 새로운 전략을 만들고 기존 전략을 개선하는, 완전한 인간-AI 공생 워크플로우가 구현됨.

결론: 최종 방향성의 확립

본 'AI 트레이딩 어시스턴트 V6.0: 최종 시스템 설계도'는 프로젝트의 방향성에 대한 모든 논의를 종결하고, 우리가 나아가야 할 길을 명확히 제시하는 최종 선언문이다. 이 설계도는 단순한 기능 목록을 넘어, 인간의 전문성과 AI의 분석력이 어떻게 시너지를 이루어 함께 성장하는 '살아있는 시스템'을 만들 수 있는지에 대한 구체적인 청사진을 담고 있다.

이전 기획안의 복잡성은 제거된 것이 아니라, 프로덕션 등급의 안정성과 지속 가능한 성장을 위해 반드시 필요한 논리적 구조로 재편성되었다. 10개의 데이터베이스와 5개의 아키텍처 계층은 이제 각자의 명확한 역할과 상호작용을 통해 하나의 유기적인 '지식 공장'을 형성한다.

이 문서를 우리의 최종 '헌법'으로 삼음으로써, 개발팀은 더 이상 방향성에 대한 고민 없이 오직 구현에만 집중할 수 있을 것이다. 우리는 이 설계를 바탕으로 흔들림 없이 코드를 작성하고, 시장에서 지속 가능한 지적 우위(Intellectual Edge)를 복리처럼 쌓아나가는 강력한 AI 트레이딩 파트너를 현실로 만들어 나갈 것이다. 이것이 우리가 진정으로 가야 할 방향이다.

참고 자료

1. AI Trading Assistant V5.5 - Production Ready System 완전 기획안
2. medium.com, 8월 1, 2025에 액세스,
<https://medium.com/@learnbackend/what-is-a-production-ready-software-f816f3b9b9ef#:~:text=With%20that%20being%20said%2C%20we,is%20tested%2C%20and%20is%20documented.>
3. What Is "Production-Grade" Software? | Ready, Set, Cloud!, 8월 1, 2025에 액세스,
<https://readyssetcloud.io/blog/allen.helton/what-is-production-grade-software>
4. What is a Production-Ready Software? | by Learn Backend - Medium, 8월 1, 2025에 액세스,
<https://medium.com/@learnbackend/what-is-a-production-ready-software-f816f3b9b9ef>
5. Websocket Streaming - Alpaca API Docs, 8월 1, 2025에 액세스,
<https://docs.alpaca.markets/docs/websocket-streaming>
6. Alpaca Trading API Guide - A Step-by-step Guide - AlgoTrading101 Blog, 8월 1, 2025에 액세스,
<https://algotrading101.com/learn/alpaca-trading-api-guide/>
7. WebSocket Stream - Alpaca API Docs, 8월 1, 2025에 액세스,
<https://docs.alpaca.markets/docs/streaming-market-data>
8. Advice on real-time API or WebSocket : r/algotrading - Reddit, 8월 1, 2025에 액세스,
https://www.reddit.com/r/algotrading/comments/1alcy0j/advice_on_realtime_api_

[or websocket/](#)

9. Alpaca Market Data API (Part 1) - Streaming with Python and Websockets - YouTube, 8월 1, 2025에 액세스, https://www.youtube.com/watch?v=Mv6c_9FqNx4
10. Real-time Crypto Trading System with Alpaca API and Transformer Model | by Rich Tsai, 8월 1, 2025에 액세스, <https://medium.com/@rich.tsai1103/real-time-crypto-trading-system-with-alpaca-api-and-transformer-model-b8bf1df42f36>
11. The Future of Backtesting: A Deep Dive into Walk Forward Analysis - Interactive Brokers LLC, 8월 1, 2025에 액세스, <https://www.interactivebrokers.com/campus/ibkr-quant-news/the-future-of-backtesting-a-deep-dive-into-walk-forward-analysis/>
12. The Future of Backtesting: A Deep Dive into Walk Forward Analysis - PyQuant News, 8월 1, 2025에 액세스, <https://www.pyquantnews.com/free-python-resources/the-future-of-backtesting-a-deep-dive-into-walk-forward-analysis>
13. Walk-Forward Optimization: How It Works, Its Limitations, and Backtesting Implementation, 8월 1, 2025에 액세스, <https://blog.quantinsti.com/walk-forward-optimization-introduction/>
14. About the Walk-Forward Optimizer - the TradeStation Platform, 8월 1, 2025에 액세스, https://help.tradestation.com/09_01/tswfo/topics/about_wfo.htm
15. What is a Walk-Forward Optimization and How to Run It? - Algo Trading 101, 8월 1, 2025에 액세스, <https://algotrading101.com/learn/walk-forward-optimization/>
16. Logging | Apps Script - Google for Developers, 8월 1, 2025에 액세스, <https://developers.google.com/apps-script/guides/logging>
17. Extending Google Sheets | Apps Script, 8월 1, 2025에 액세스, <https://developers.google.com/apps-script/guides/sheets>
18. Change log to track status using Apps Script in Google Sheet, 8월 1, 2025에 액세스, <https://support.google.com/docs/thread/110880157/change-log-to-track-status-using-apps-script-in-google-sheet?hl=en>
19. Automated Cell Tracking in Google Sheets using Google Apps Script - Code Walkthrough, 8월 1, 2025에 액세스, <https://www.youtube.com/watch?v=ixl1FTkgtOQ>