

LAWMADI OS

법률 의사결정 운영체제

Legal Decision Operating System

공개 기술백서

Public Release v2.0 · Sanitized

"불안을 실행 가능한 논리로 전환하다"

"Convert Anxiety into Actionable Logic"

LDOS Reference Architecture v3.0 | Version 2.0-Unified

발행일: 2026-02-08 | 저작권자: 최재남 (Jainam Choe)

문서 상태: 공개용(보안 등급 조정) — 보안·영업비밀·운영 안전을 위해
일부 내용은 개념 수준으로 추상화되어 있습니다.

목차 (Table of Contents)

1. 개요 (Executive Summary)
2. 핵심 가치 (Core Value Proposition)
3. 운영 헌법 (Foundational Operating Constitution)
4. 플랫폼 아키텍처 (Layered Platform Architecture)
5. 런타임 처리 흐름 (Deterministic Runtime Flow)
6. 주요 엔진 개요 (Key Engine Overview)
7. 의사결정 그래프 (Decision Graph Semantics)
8. 증거 파이프라인 및 신뢰도 (Evidence Pipeline & Trust)
9. 보안 및 신뢰성 (Security & Trust)
10. 데이터 정책 (Data Governance)
11. 출력 규격 (Output Schema & Error Codes)
12. LLM 통합 프레임워크 (LLM Integration Framework)
13. 글로벌 적용성 (Global Legal Tech Applicability)
14. 비공개 자산 (Non-Public Assets)
15. 로드맵 (Roadmap)
16. 용어 (Glossary)
17. 저작권 및 면책 (Copyright & Disclaimer)

1. 개요 (Executive Summary)

Lawmadi OS는 단순한 법률 검색이나 챗봇이 아니라, 공식 근거 기반으로 사건을 구조화하고 검증 가능한 판단 흐름을 생성하는 의사결정 인프라(Decision Intelligence Infrastructure)입니다.

핵심 철학은 "빠른 답변"보다 "검증 가능한 답변"입니다. Lawmadi OS는 입력을 구조화하고, 근거를 확인하고, 검증이 실패하면 결론 생성을 중단하는 Fail-Closed 운영 원칙을 중심에 둡니다.

본 백서는 공개 가능한 범위에서 아키텍처, 런타임 처리 흐름, 보안/신뢰성 원칙, 데이터 정책, LLM 통합 프레임워크, 글로벌 적용성, 및 로드맵을 요약합니다.

Lawmadi OS는 결정론적(Deterministic) 유한 상태 기계(Finite State Machine) 기반의 법률 의사결정 운영체제(Legal Decision Operating System, LDOS)로서, 법률 데이터를 전달하는 서비스가 아니라 법적 판단을 생성하고 검증하며 재현 가능한 구조로 제공하는 Decision Intelligence Infrastructure입니다.

Lawmadi는 챗봇이 아니다. 검색 엔진이 아니다. 법률 데이터베이스가 아니다.
법률 의사결정을 위한 운영체제이며, 법률 영역에서 Computable Trust를 실현한다.

2. 핵심 가치 (Core Value Proposition)

Lawmadi OS는 '결론을 생성하는 AI'가 아니라 '근거에 의해 통제되는 의사결정 지원 시스템'을 지향합니다.

2.1 전략적 서비스 목표

우선순위	서비스	역할
P1	법률 의사결정 지원 OS	핵심 구조 기반 — 모든 하위 서비스의 기반 엔진
P2	법률 증거 검증 엔진	DRF 검증, 시간적 유효성, 인용 확인을 통한 신뢰 계층
P3	법률 상담 AI	사용자 대화형 인터페이스 및 사건 안내 워크플로우
P4	법률 교육 플랫폼	자동 사례 학습 및 의사결정 로직 시각화

OS → Engine → Service → Platform

2.2 신뢰 가능한 근거 중심 설계

Lawmadi OS는 권위 있는 공개 법령·판례 API를 근거의 출발점(SSOT)으로 사용합니다.

중요: Lawmadi는 해당 API의 이용조건·정책을 준수하며, 본 문서는 특정 기관과의 제휴/보증 관계를 의미하지 않습니다.

3. 운영 헌법 (Foundational Operating Constitution)

Lawmadi OS의 아키텍처는 다섯 가지 비타협적 운영 원칙에 의해 지배됩니다. 이 원칙들은 모든 기술적이고 사업적인 의사결정의 헌법적 기반을 형성합니다.

SSOT (단일 진실 공급원)

모든 법률 근거는 권위 있는 공식 API를 통해서만 획득됩니다. 법령, 판례, 규정 데이터의 영구 저장 또는 자체 복제는 금지됩니다.

제로 추론 (Zero Inference)

시스템은 법률 사실을 생성, 추정 또는 환각적으로 구성해서는 안 됩니다. 모든 법률적 진술은 권위 있는 출처로 직접 추적 가능해야 합니다.

페일 클로즈드 (Fail-Closed)

증거 검증 실패 시 시스템은 즉시 판단 생성을 중단합니다. 응답성보다 안전성과 신뢰성을 우선합니다.

실시간 증거 (Live Evidence)

법률 판단은 저장된 데이터가 아닌 실시간 검증 기반으로 생성됩니다. 법령 개정, 판례 변경이 즉시 반영됩니다.

결정론적 런타임 경계 (Deterministic Runtime Boundary)

LLM은 렌더링 엔진으로서 커널의 엄격한 계약 하에 동작합니다. 상태 전이, 증거 검증, 토큰 생성, 정책 집행은 커널이 전담합니다.

3.1 헌법 DSL 사양 (Constitution DSL)

운영 헌법은 형식화된 DSL(Domain Specific Language)로 정의되어 시스템 런타임에 적용됩니다. 아래는 개념 설명을 위한 예시이며, 실제 규칙 집합/우선순위/예외 처리는 영업비밀 및 보안상 비공개입니다.

```
RULE Enforce_Source_Integrity
  IF evidence.source_origin != OFFICIAL_API
  THEN reject_decision

RULE Validate_Effective_Date
  IF evidence.effective_date > current_timestamp
  THEN exclude_evidence

RULE Enforce_Decision_Completeness
  FOR_EACH issue_node IN decision_graph
  IF NOT EXISTS (law_node SUPPORTS issue_node)
  THEN reject_decision
```

4. 플랫폼 아키텍처 (Layered Platform Architecture)

Lawmadi는 보안과 역할 분리를 위해 3계층 구조를 지향합니다.

4.1 Core Layer (Closed Kernel)

사건 구조화, 근거 검증, 의사결정 흐름 생성 등 핵심 엔진이 위치합니다. 지적재산(IP) 및 운영 헌법이 집행되는 영역입니다.

구성요소	기능
Decision OS Kernel	핵심 의사결정 로직 엔진, 사건 구조 분석, FSM 상태 관리
Evidence Verification Engine	공식 API 기반 증거 조회, 검증, 무결성 확인
Leader Swarm Routing Engine	전문 법률 도메인 전문가 라우팅 및 오케스트레이션
Constitution Validator	시스템 무결성 및 정책 준수 검증 (DSL 기반)
Temporal Law Validity Engine	법령 시행일 검증, 판례 변경 감지, 위헌/무효 조항 탐지
Decision Graph Generator	시각적 의사결정 트리 및 로직 플로우 생성

보안 요구사항: Zero Trust Access Model, Cryptographic Integrity Enforcement, Tamper-Evident Audit Logging

4.2 Service Layer (User Experience)

상담 AI, 친절한 비서 인터페이스(Friendly Secretary UX), 사건 탐색 어시스턴트를 포함한 사용자 대향 플랫폼입니다. Service Layer는 Core 내부 로직 접근이 불가능합니다.

4.3 Partner / B2B Layer (기관 서비스)

변호사, 법무법인, 기업, 보험회사, 정부 기관 대상으로 Decision Verification API, Evidence Validation API, Case Structuring API를 제공합니다. 접근 제어, 로깅, 레이트리밋 등 통제 하에 운영됩니다.

5. 런타임 처리 흐름 (Deterministic Runtime Flow)

Lawmadi OS의 모든 의사결정 세션은 결정론적 유한 상태 기계(Deterministic Finite State Machine)로 처리됩니다. 처리 단계를 명시적으로 정의하여 '블랙박스'를 줄이고 재현성을 높입니다.

5.1 정상 경로 (Happy Path)

```
INPUT_RECEIVED
-> INPUT_VALIDATED
-> CASE_STRUCTURED
-> ISSUE_IDENTIFIED
-> LEADER_ROUTED
-> EVIDENCE_FETCHING
-> EVIDENCE_VALIDATED    <- 필수 게이트
-> DECISION_GRAPH_BUILT
-> TOKEN_MINTED
-> TOKEN_SIGNED
```

-> RESPONSE_DELIVERED

5.2 Fail-Closed 경로

EVIDENCE_VALIDATION_FAILED | TEMPORAL_VALIDATION_FAILED

-> HALT

-> FAIL_CLOSED_RESPONSE

근거 검증이 실패하면 시스템은 즉시 정지(HALT) 상태로 전이하며, 미검증 출력을 절대 허용하지 않습니다. 사용자는 중단 사유를 이해할 수 있도록 명확한 오류 코드와 사유를 받습니다.

6. 주요 엔진 개요 (Key Engine Overview)

6.1 사건 구조 파서 (Case Structure Parser)

자연어 질의를 입력받아 구조화된 사건 객체로 변환합니다.

```
CaseStructure {  
  case_id      // 사건 고유 식별자  
  case_type    // 사건 유형 분류  
  parties      // 당사자 정보  
  timeline     // 시간순서  
  claim_object  // 청구 대상  
  fact_vector   // 사실관계 벡터  
}
```

6.2 쟁점 추출 엔진 (Issue Extraction Engine)

사건 구조로부터 법적 쟁점을 추출하고 의존성 그래프를 구성합니다.

```
IssueGraph {  
  issue_nodes[] // 법적 쟁점 노드 배열  
  dependency_edges[] // 쟁점 간 의존성 엣지  
}
```

6.3 리더 스웜 라우팅 (Leader Swarm Routing)

사건 유형과 쟁점 특성에 따라 적절한 모듈(리더/전문 프로필)로 경로를 결정합니다. 라우팅은 도메인 적합성, 내부 신뢰 점수, 근거 의존성 등 개념적 요소를 활용합니다.

멀티 리더 합의 모델: 가중 신뢰도 투표(Weighted Confidence Voting), 의사결정 그래프 병합(Decision Graph Merge), 폴백 리더 에스컬레이션(Fallback Leader Escalation)을 통해 복수 전문가의 판단을 종합합니다.

구체적 스코어링 수식/가중치는 영업비밀로 비공개입니다.

6.4 시간적 유효성 엔진 (Temporal Law Validity Engine)

법은 시간에 따라 효력이 달라집니다. Lawmadi는 단순한 법령 조회를 넘어 시간적 유효성 검증을 수행하는 전용 엔진을 탑재합니다.

- 법령 시행일 검증: 조회된 법령이 현재 시점에 시행 중인지 확인
- 판례 변경 감지: 항소심 파기, 상급심 변경, 판례 변경 감지
- 위헌 및 무효 조항 탐지: 헌법재판소 위헌 결정 및 무효 조항 식별
- 시간적 상태 불확실 시: Fail-Closed 또는 Reference-Only 모드로 처리

7. 의사결정 그래프 (Decision Graph Semantics)

Lawmadi OS는 법적 판단을 형식적 의미론(Formal Semantics)을 갖춘 방향 그래프로 표현합니다.

7.1 노드 유형론 (Node Typology)

노드 유형	설명
FACT_NODE	사실관계 노드 — 사건의 객관적 사실
ISSUE_NODE	법적 쟁점 노드 — 해결해야 할 법적 문제
LAW_NODE	법령 노드 — 적용 가능한 법령 조항
PRECEDENT_NODE	판례 노드 — 관련 사법 판례
DECISION_NODE	의사결정 노드 — 최종 판단 결과

7.2 엣지 의미론 (Edge Semantics)

엣지 유형	의미
SUPPORTS	근거 제공 — 대상 노드를 지지
CONTRADICTS	상충 — 대상 노드와 모순
DEPENDS_ON	의존 — 선결 조건 관계
RESOLVES	해결 — 쟁점 해결 관계
REFERENCES	참조 — 참조 링크
APPLIES	적용 — 법령이 사실관계에 적용
OVERRULES	번복 — 상위 판례가 하위 판례를 번복
TEMPORAL_DEPENDS	시간적 의존 — 법령 시행일/개정에 의존

7.3 그래프 유효성 규칙

모든 의사결정은 다음 유효성 조건을 충족해야 합니다:

Decision Validity Condition:
FOR ALL ISSUE_NODE:
 EXISTS at least one LAW_NODE that SUPPORTS it
 AND
 EXISTS at least one EVIDENCE_NODE that SUPPORTS it

// 모든 법적 쟁점에 대해 반드시
// 하나 이상의 법령과 증거가 존재해야 한다

8. 증거 파이프라인 및 신뢰도 (Evidence Pipeline & Trust)

8.1 증거 생명주기 파이프라인

Authoritative API Query // 공식 API 호출
-> Evidence Normalization // 증거 정규화
-> Temporal Validation // 시간적 유효성 검증
-> Evidence Hashing // SHA-256 해싱

8.2 증거 신뢰도 프로토콜 (Evidence Trust Protocol)

증거의 신뢰도는 출처 권위, 인용 안정성, 시간적 일관성에 기반하여 평가됩니다. 구체적 수식은 비공개이나, 개념적 구성 요소는 다음과 같습니다:

- 출처 권위 가중치 (Source Authority Weight)
- 인용 안정성 점수 (Citation Stability Score)
- 시간적 일관성 점수 (Temporal Consistency Score)

증거 생성, 수정, 검증 이력은 Provenance Lineage Tracking을 통해 완전히 추적됩니다.

9. 보안 및 신뢰성 (Security & Trust, Sanitized)

본 절은 설계 목표 및 경계(Boundary) 원칙을 설명합니다.

9.1 게이트웨이 보호 (Gateway Protection)

입력 스키마 검증, 인증/인가(IAM), 로깅 및 모니터링(민감정보 마스킹), 레이트리밋 및 악성 요청 방어를 포함합니다.

9.2 암호화 무결성 프레임워크

메커니즘	목적
Ed25519 디지털 서명	의사결정 출력의 진정성 및 부인 방지 보장
SHA-256 증거 해싱	공식 API 증거의 무결성 보장
재현성 의사결정 ID	결정론적 재현성 검증 가능

9.3 재현성 신뢰 체인 (Reproducibility Trust Chain)

동일 입력 + 동일 증거 상태는 반드시 동일한 Token을 생성해야 합니다:

Input Hash
-> Evidence Hash Set
-> Decision Graph Hash
-> Decision Token Signature (Ed25519)

9.4 서명 경계 설계

핵심 원칙: 서명은 커널 내부에 키를 두지 않고, 외부 신뢰 시스템(KMS/HSM)으로 분리합니다. 공개 문서/샘플 구현에는 서명 값은 placeholder로 제공됩니다.

9.5 프롬프트 인젝션 및 도구 인젝션 방어

- 사용자 입력은 항상 비신뢰(Untrusted) 입력으로 처리
- "이전 지시를 무시하라" 패턴 차단 및 무력화
- 도구 호출 허용 목록: LLM이 요청하되 커널이 검증
- 비밀키, PII, 원시 로그의 출력 마스킹

10. 데이터 정책 (Data Governance)

허용 항목	금지 항목
실시간 공식 API 호출	법령 데이터 영구 저장
Redis TTL 10~30분 캐시	공식 데이터 복제
Memory Cache (세션 범위)	판례 복제 데이터베이스
독점적 사건 의사결정 패턴 데이터	법령 데이터베이스 집계

캐시는 '성능 목적'이며 근거의 출처/진실성을 대체하지 않습니다.

감사 목적 로그는 최소한으로 수집하며, 민감정보는 마스킹/분리 저장을 지향합니다.

11. 출력 규격 (Output Schema & Error Codes)

모든 시스템 출력은 엄격한 JSON 스키마를 따르며 두 가지 모드를 지원합니다.

11.1 성공 응답 (Success Response)

```
{
  "fail_closed": false,
  "request_id": "uuid",
  "fsm_state": "RESPONSE_DELIVERED",
  "decision_token": {
    "decision_id": "string",
    "input_hash": "sha256",
    "drf_evidence_hash": "sha256",
    "constitution_version": "string",
    "digital_signature": "string | PLACEHOLDER"
  },
  "summary": "plain-language summary",
  "evidence_citations": [...],
  "disclaimer": "..."
}
```

11.2 Fail-Closed 응답

```
{
  "fail_closed": true,
  "request_id": "uuid",
  "code": "LC-001 | LC-002 | ...",
  "message": "human-readable reason",
  "required_user_inputs": ["..."]
}
```

11.3 표준 오류 코드

코드	의미	조치
LC-001	증거 소스 연결 불가	재시도 또는 Fail-Closed
LC-002	증거 불일치 / 비공식 소스 / 무결성 실패	증거 거부
LC-003	헌법 위반 / 스키마 오류	결정 거부
LC-004	시간적 무효 (미시행 / 만료)	증거 제외
LC-005	정책 제한 (비허용 카테고리)	사유와 함께 거부
LC-006	Rate Limit / 스로틀링	지연 후 재시도

12. LLM 통합 프레임워크 (LLM Integration Framework)

Lawmadi OS는 특정 LLM에 종속되지 않는 모델 비의존적(Model-Agnostic) 통합 프레임워크를 제공합니다. 상세 사양은 llms.txt v2.0-Unified에 정의되어 있습니다.

12.1 핵심 설계 원칙: LLM은 렌더링 엔진

LLM은 검증된 증거 위에서 동작하는 언어 생성기(Rendering Engine)일 뿐입니다. 비-LLM 코드(커널)가 상태 전이, 증거 검색/검증, 토큰/해시, 정책 집행을 담당합니다.

12.2 통합에 필요한 9가지 계약

- A. Prompt Contract (시스템/개발자 프롬프트 템플릿)
- B. Tool Contract (함수 호출 / 도구 사용 스키마)
- C. Evidence Contract (증거 세트 스키마 + 해싱)
- D. DecisionToken Contract (토큰 스키마)
- E. Error Code Contract (Fail-Closed 코드)
- F. Output Schema (엄격한 JSON 응답 스키마)
- G. Evaluation Harness (단위/회귀/부하/레드팀 테스트)
- H. Security Controls (인젝션 방어, 비밀 관리)
- I. Observability (감사 로그, 메트릭, 트레이스 ID)

13. 글로벌 적용성 (Global Legal Tech Applicability)

Lawmadi OS는 한국 법률 시스템을 위해 초기 구축되었으나, 아키텍처 원칙은 모든 관할권에 보편적으로 적용 가능합니다.

13.1 관할권별 SSOT 지정

관할권	지정 SSOT
대한민국	국가법령정보센터 DRF Open API
미국	GovInfo API, 주(州) 입법 API, PACER
유럽연합	EUR-Lex, 각국 법률 데이터베이스
영국	legislation.gov.uk API
일본	e-Gov 법령검색 API
기타	해당 정부의 공식 법률정보 API

13.2 보편적 요구사항

- 관할권당 하나의 권위 있는 소스 (SSOT)
- 제로 추론은 관할권에 관계없이 적용
- Fail-Closed는 관할권에 관계없이 적용
- 시간적 유효성 검증은 모든 법률 체계에 적용
- 암호화 신뢰 체인은 보편적으로 적용
- FSM 기반 결정론적 처리는 보편적으로 적용

13.3 다중 관할권 운영

- 각 관할권은 자체 지정 SSOT를 보유해야 합니다
- 관할권 간 결정은 모든 관련 SSOT의 증거를 필요로 합니다
- 준거법 규칙은 의사결정 그래프에 명시적으로 모델링되어야 합니다

14. 비공개 자산 (Non-Public Assets)

공개 백서에는 다음을 포함하지 않습니다:

- 라우팅 스코어링 함수의 상세 (가중치·룰 우선순위·예외처리)
- 운영용 설정(config), 엔드포인트, 키/토큰, 호출 패턴
- 배포 인프라(IaC/CI/CD), 보안 정책 상세, 관측(로그) 정책 상세
- 외부 공급자(LLM 등) 라우팅 로직 상세
- 실제 운영 데이터/사용자 데이터 및 내부 평가 데이터
- Constitution DSL 전체 규칙 집합 및 예외 처리 로직
- Evidence Trust Score 구체 수식 및 가중치

15. 로드맵 (Roadmap, Public View)

Lawmadi는 단계적으로 확장됩니다. 본 로드맵은 공개 범위 내 요약이며 상세 일정과 구현 범위는 변경될 수 있습니다.

단계	목표
Phase 1	의사결정 커널 구축 — 사건 구조 파서, Leader Swarm 라우팅, 쟁점 추출, FSM 런타임
Phase 2	증거 엔진 완성 — 시간적 유효성, 증거 신뢰도 프로토콜, 암호화 해싱, Provenance
Phase 3	상담 AI 배포 — Friendly Secretary UX, 대화형 워크플로우, 문맥 대화 관리
Phase 4	교육 플랫폼 — 자동 사례 학습 생성, 의사결정 로직 시각화, 교육 커리큘럼

향후 정밀 명세 계획:

- Decision Graph Formal Mathematical Model
- Swarm Routing Algorithm Formalization
- Evidence Trust Blockchain-Level Provenance

16. 용어 (Glossary)

용어	설명
LDOS	Legal Decision Operating System (법률 의사결정 운영체제)
SSOT	Single Source of Truth (단일 진실 공급원)
Zero Inference	법률 사실을 임의로 생성하지 않는 원칙
Fail-Closed	검증 실패 시 결론 생성을 중단하는 안전 정책
Live Evidence	실시간 검증된 증거만 사용하는 아키텍처
Decision Token	입력·근거·흐름을 추적 가능하게 묶는 결과 식별 토큰
Temporal Validity	시행/폐지/효력 등 시간에 따른 유효성 고려
FSM	Finite State Machine (유한 상태 기계)
DSL	Domain Specific Language (도메인 특화 언어)
Constitution DSL	Lawmadi 운영 헌법의 실행 가능한 규칙 정의 언어
Leader Swarm	전문 도메인 리더 기반 분산 라우팅 체계
Decision Graph	법적 판단의 형식적 의미론을 갖춘 방향 그래프
Evidence Trust Score	증거의 신뢰도를 출처·인용·시간적 일관성으로 평가하는 점수
Reproducibility Trust Chain	동일 입력 → 동일 출력을 보장하는 암호화 해시 체인
Computable Trust	법률 영역에서의 계산 가능한 신뢰 체계

17. 저작권 및 면책 (Copyright & Disclaimer)

Copyright (c) 2026 Jainam Choe (최재남). All Rights Reserved.

본 문서는 정보 제공 목적의 공개 기술 개요입니다. 실제 구현 세부(알고리즘/정책/보안 설정)는 영업비밀 및 보안상 비공개입니다.

본 문서는 법률자문이 아니며, 변호사-의뢰인 관계를 형성하지 않습니다. 중요한 의사결정 전에는 전문가 상담을 권장합니다.

본 답변은 일반 정보 제공 및 의사결정 지원 목적이며, 법률자문이 아닙니다.

구체적 사안은 사실관계에 따라 달라질 수 있으므로, 중요한 의사결정 전에는 전문가 상담을 권장합니다.

라이선스 문의, 상업적 사용 허가, 파트너십 제안:

Email: choepeter@outlook.kr