



INTEGRATION_CONTRACT.md

Mental Health Ontology Knowledge Graph Project

1. Purpose of This Document

This document defines the **explicit integration agreements** between all system components and team members.

Its purpose is to ensure that:

- each module communicates using **agreed formats**
- no component performs responsibilities outside its role
- symbolic reasoning remains **sound, explainable, and auditable**
- the system behaves consistently across development stages

This contract is **binding** for all contributors.

2. System-Wide Integration Principles

All integrations must follow these principles:

1. **Ontology is the single source of truth**
 2. **Reasoning is symbolic and rule-based**
 3. **Statistical components never trigger inference**
 4. **Safety overrides all reasoning**
 5. **Every inference must be explainable**
 6. **All components must be inspectable**
-

3. Component Ownership Summary

Component	Owner
Ontology & Base KG	Person 1

Component	Owner
Graph Manager	Person 1
NLP Extraction & Mapping	Person 2
Evidence Confidence	Person 2
SWRL Reasoning	Person 3
SPARQL Materialization	Person 3
Causal Explanations (WHY)	Person 3
Ranking & Uncertainty	Person 3
Safety & Escalation	Person 3
Audit Trail	Person 3

No component may exceed its ownership boundary.

4. Ontology & Knowledge Graph Contract (Person 1 → All)

Ontology Guarantees

Person 1 guarantees that:

- All ontology URIs are **stable after freeze**
- Class hierarchy is consistent
- Property domains and ranges are defined
- No clinical diagnosis axioms exist

Mandatory Ontology Classes

- Emotion
- Symptom
- Trigger
- MentalState
- RiskLevel
- Intervention
- Student

Mandatory Object Properties

- experiencesEmotion
 - hasSymptom
 - triggeredBy
 - associatedWith
 - increasesRiskOf
 - recommendedIntervention
-

5. NLP Evidence Contract (Person 2 → Graph Manager)

Evidence Output Format

Every extracted concept must follow this structure:

```
{  
  "uri": "mh:Insomnia",  
  "label": "Insomnia",  
  "class": "Symptom",  
  "confidence": 0.92,  
  "method": "keyword | semantic",  
  "surfaceForm": "can't sleep",  
  "turnIndex": 3,  
  "timestamp": "ISO-8601"  
}
```

Session Evidence Package

```
{  
  "session_id": "uuid",  
  "evidence": [ ... ],  
  "confidence_summary": {  
    "average": 0.78,  
    "min": 0.61,  
    "max": 0.93  
  }  
}
```

Hard Constraints

- No inference or aggregation
 - No severity classification
 - No risk labels
 - No escalation detection
-

6. Graph Manager Contract (Person 1 ↔ Person 3)

Graph Manager Responsibilities

- Load ontology & base KG
- Create per-session subgraph
- Insert evidence triples
- Maintain persistence metadata
- Expose query interface

Evidence → Triple Mapping

Example:

```
:Student_123 mh:hasSymptom mh:Insomnia .  
:Student_123 mh:evidenceConfidence "0.92" .
```

Persistence Metadata

- turnIndex
 - occurrenceCount
 - lastSeenTimestamp
-

7. Reasoning Contract (Person 3)

SWRL Rules

- Must only reference ontology-defined classes & properties
- Must have:

- rule ID
- description
- intent
- priority

Example:

R_Anxiety_01:

Insomnia \wedge Stress \wedge ExamPressure \rightarrow AnxietyRisk

SPARQL Usage Rules

- CONSTRUCT / INSERT: materialization only
 - SELECT: explanation retrieval
 - No logical inference via SPARQL
-

8. Explanation Contract (Person 3 → API / UI)

Explanation Schema

```
{  
  "riskState": "AnxietyRisk",  
  "confidence": 0.72,  
  "rulesFired": ["R_Anxiety_01"],  
  "evidence": [...],  
  "causalChain": [  
    "Insomnia",  
    "Stress",  
    "AcademicStress",  
    "AnxietyRisk"  
,  
    "uncertaintyDrivers": [  
      "short duration",  
      "limited repetition"  
    ]  
}
```

Explanation Guarantees

- Every inference has a WHY
 - WHY is derived from KG traversal
 - No hardcoded text explanations
-

9. Ranking & Recommendation Contract

Ranking Inputs

- rule priority
- evidence count
- persistence
- ontology `causalStrength` (annotation only)

Ranking Constraints

- Ranking does not trigger inference
 - Ranking never overrides safety
 - Ranking is deterministic
-

10. Safety & Escalation Contract

Hard Escalation

Triggered when:

- self-harm
- suicidal ideation

Actions:

- bypass reasoning
 - return emergency response
 - log escalation
-

Soft Escalation

Triggered when:

- multiple high-risk states
- strong persistence

Actions:

- suggest professional help
 - preserve user autonomy
-

Escalation Log Schema

```
{  
  "type": "hard | soft",  
  "trigger": "phrase or pattern",  
  "timestamp": "ISO-8601",  
  "bypassedReasoning": true  
}
```

11. Audit Trail Contract

Each session must log:

- evidence inserted
- rules fired
- explanations generated
- escalations triggered

Logs must be:

- human-readable
 - exportable
 - immutable per session
-

12. Change Management & Freeze Rules

Ontology Freeze

Once frozen:

- no class/property changes

- no URI renaming

Rule Freeze

Once validated:

- no semantic rule changes
 - only documentation updates allowed
-

13. Conflict Resolution

If conflicts arise:

1. Refer to `context.md`
 2. Refer to this contract
 3. Ontology correctness overrides implementation convenience
 4. Safety overrides all logic
-

14. Final Agreement

All team members agree that:

- This contract governs integration
 - Violations must be fixed immediately
 - No silent assumptions are allowed
-

End of Document
