Decision Agent - Risk Fusion & Alert Generation

Overview

The Decision Agent is the final layer of the multi-agent system. It fuses outputs from all three upstream agents (Sensor, Patient, Clinical) to calculate a global risk score and make intelligent alert decisions.

Core Responsibility

"Should we bother a human?"

The Decision Agent answers this by:

1. Combining all risk signals
2. Requiring persistence (no single-spike alerts)
3. Managing alert cooldowns (prevent fatigue)
4. Generating explainable decisions

Fusion Formula

```
global_risk = α × physio_risk × sensor_confidence
            + β × normalized_sensitivity × patient_confidence
            + γ × clinical_risk × clinical_confidence

Where:
α = 0.5  (50% weight on real-time physiology)
β = 0.3  (30% weight on personal context)
γ = 0.2  (20% weight on static medical history)
```

Component Breakdown

Sensor Component (50%):

- `physio_risk`: Arrhythmia + HRV irregularity score (0-1)
- `sensor_confidence`: ECG signal quality (0-1)
- Real-time, changes every 2 seconds

Patient Component (30%):

- `sensitivity_factor`: 0.8-1.3 mapped to 0-1
- `patient_confidence`: Days tracked / 30 (0-1)
- Personal context, adapts daily

Clinical Component (20%):

- `clinical_risk`: Logistic regression output (0-1)
- `clinical_confidence`: Always 1.0 (static data)
- Baseline vulnerability, changes manually

Decision Logic

Three Decision Levels

1. **NO_ALERT** (Green)
    - Global risk < 0.5
    - Everything within normal range
    - Continue monitoring
2. **MONITOR** (Orange)
    - Global risk ≥ 0.5 for 3+ consecutive readings
    - Elevated but not critical
    - Watch for escalation
3. **ALERT** (Red)
    - Global risk ≥ 0.7 for 5+ consecutive readings
    - Critical risk sustained over time
    - Action required

Persistence Requirements

Why? Prevent false alarms from single-spike measurements.

- **Monitor**: Requires 3 consecutive readings above 0.5
- **Alert**: Requires 5 consecutive readings above 0.7

```
Reading:  0.6  0.7  0.6  0.8  0.75  0.78  0.72  0.74
Counter:  1    2    0    1    2     3     4     5
Decision: NO   NO   NO   NO   NO    MONITOR MONITOR ALERT
```

Alert Cooldown

Why? Prevent alert fatigue and spam.

- **Duration**: 30 minutes after each alert
- **Behavior**: System continues monitoring but won't trigger new alerts
- **Display**: Shows countdown timer in UI

```
Time:    12:00  12:10  12:20  12:30  12:35
Risk:    0.8    0.85   0.82   0.78   0.76
Alert:   ✓      ✗      ✗      ✗      Ready
         ALERT  COOLDOWN COOLDOWN COOLDOWN  Can alert again
```

Explanation Generation

Every decision must be explainable. The system answers:

1. What changed?

- Risk level: CRITICAL (>85%) or HIGH (70-85%)
- Global risk percentage

2. Compared to what baseline?

- Patient's personal norms
- Clinical risk factors

3. For how long?

- Number of consecutive high-risk readings
- Minimum required for alert

4. Why now?

- Specific contributors (arrhythmia, behavioral state, clinical factors)
- Protective factors if any

Example Alert Explanation

🚨 HIGH RISK DETECTED (78%)

Contributors:
- Physiological: arrhythmia probability 65%, irregular rhythm detected
- Behavioral: Patient state is DEGRADED (sensitivity 1.15x normal)
- Clinical: High baseline risk (60%) from medical history

Sustained for 5 consecutive readings (minimum 5 required)

⚠️ Patient has shown concerning behavioral patterns over recent days

📞 RECOMMENDATION: Contact healthcare provider or seek immediate evaluation

API Endpoints

GET /api/decision/current

Current decision status:

json


```
{
  "consecutive_monitor": 2,
  "consecutive_alert": 0,
  "in_cooldown": false,
  "time_until_ready_minutes": null,
  "total_alerts": 3,
  "total_monitors": 15,
  "last_decision": {
    "global_risk": 0.52,
    "decision": "MONITOR",
    "explanation": "...",
    "components": { ... }
  },
  "alert_history_count": 3
}
```


GET /api/decision/alerts?limit=10

Alert history:

```json
{
 "alerts": [
  {
   "alert_id": 1,
   "timestamp": "2025-12-27T10:30:00",
   "global_risk": 0.78,
   "decision": "ALERT",
   "explanation": "...",
   "components": {
    "sensor_risk": 0.65,
    "patient_sensitivity": 1.15,
    "clinical_risk": 0.60
   },
   "acknowledged": false
  }
 ]
}
```

POST /api/decision/acknowledge/{alert_id}

Mark alert as acknowledged:

```json
{
 "success": true
}
```

GET /api/decision/trend

Recent risk trend for visualization:

```json
{
 "trend": [
  { "timestamp": "2025-12-27T10:25:00", "risk": 0.45 },
  { "timestamp": "2025-12-27T10:27:00", "risk": 0.52 },
  { "timestamp": "2025-12-27T10:29:00", "risk": 0.58 }
 ]
}
```

Integration Flow

```
Every 2 seconds:

Sensor Agent outputs → physio_risk, confidence
  ↓
Patient Agent state → sensitivity_factor, confidence
  ↓
Clinical Agent state → clinical_risk, confidence
  ↓
Decision Agent fuses inputs
  ↓
Calculates global_risk
  ↓
Checks persistence counters
  ↓
Checks cooldown status
  ↓
Makes decision (NO_ALERT / MONITOR / ALERT)
  ↓
Generates explanation
  ↓
Updates UI + logs if ALERT
```

Alert Management

Alert History

- Stores last 100 alerts
- Tracks acknowledgment status
- Provides full context for each alert

Alert Acknowledgment

Users can acknowledge alerts to:

- Indicate they've reviewed the alert
- Clear visual indicators
- Track response patterns

Acknowledged alerts:

- Remain in history
- Display with reduced opacity
- Show acknowledgment timestamp

Persistent State

Stored in `data/decision_state.json`:

```json
{
  "patient_id": "patient_001",
  "created_date": "2025-12-01T00:00:00",
  "last_decision": { ... },
  "alert_history": [ ... ],
  "recent_risks": [ ... ],
  "consecutive_monitor": 2,
  "consecutive_alert": 0,
  "last_alert_time": "2025-12-27T10:30:00",
  "total_alerts": 3,
  "total_monitors": 15
}
```

Design Principles

1. Conservative by Default

- High thresholds (0.7 for alerts)
- Requires sustained elevation (5 readings)
- 30-minute cooldown between alerts

2. Explainability First

- Every decision has clear reasoning
- Specific contributors identified
- Recommendations provided

3. No Single-Spike Alerts

- Persistence requirements prevent false alarms
- Tracks consecutive readings
- Resets counter when risk drops

4. Alert Fatigue Prevention

- Cooldown period enforced
- Visual countdown in UI
- Balances vigilance with usability

5. Actionable Intelligence

- Alerts include recommendations
- Clear severity levels
- Context from all agents

Tuning Parameters

Can be adjusted based on patient population:

```python
# Fusion weights
alpha = 0.5  # Sensor weight
beta = 0.3  # Patient weight
gamma = 0.2  # Clinical weight

# Thresholds
MONITOR_THRESHOLD = 0.5  # 50%
ALERT_THRESHOLD = 0.7  # 70%

# Persistence
MONITOR_PERSISTENCE = 3  # readings
ALERT_PERSISTENCE = 5  # readings

# Cooldown
ALERT_COOLDOWN_MINUTES = 30
```

Example Scenarios

Scenario 1: Sudden Arrhythmia

```
Time: 10:00 - Patient at rest, sudden arrhythmia detected

Reading 1: physio_risk=0.65, global_risk=0.58 → NO_ALERT (needs persistence)
Reading 2: physio_risk=0.68, global_risk=0.61 → NO_ALERT (count=2)
Reading 3: physio_risk=0.72, global_risk=0.65 → MONITOR (count=3, threshold met)
Reading 4: physio_risk=0.75, global_risk=0.68 → MONITOR (count=4)
Reading 5: physio_risk=0.78, global_risk=0.72 → ALERT (count=5, alert triggered)

Result: Alert after 10 seconds of sustained elevation
```

Scenario 2: Patient in DEGRADED State

```
Patient Agent: DEGRADED state (sensitivity=1.15)

Lower sensor risk triggers alert faster:
- Normal patient needs physio_risk=0.8 for alert
- DEGRADED patient needs physio_risk=0.6 for alert

Personalization in action!
```

Scenario 3: False Alarm Prevention

```
Reading 1: global_risk=0.75 → count=1
Reading 2: global_risk=0.72 → count=2
Reading 3: global_risk=0.48 → count=0 (RESET, single spike filtered out)

No alert triggered - transient elevation ignored
```

Success Metrics

The Decision Agent is successful if:

- ✅ Detects sustained abnormalities (not transient)
- ✅ Reduces false alarm rate
- ✅ Provides clear, actionable explanations
- ✅ Adapts to individual patient context
- ✅ Prevents alert fatigue

Accuracy alone is NOT the goal - the goal is **useful, trustworthy alerts** that help patients without overwhelming them.