Task 3: Akshara Pulse – Platform Publisher + Analytics Agent

Context:

With multilingual generation, sentiment tuning, and voice output in place, the next step is to simulate social media publishing and capture user engagement metrics to inform future content creation. This will form the adaptive loop for the Sanatan AI Engine.

Objectives

1. Agent J: Platform Publisher
    - Simulate posting to 3 platforms:
        - Instagram (Text + Voice thumbnail)
        - Twitter (Short text + TTS snippet)
        - LinkedIn (Formatted post with title + summary + TTS)
    - Support preview mode (generates a JSON/post object but doesn't post)
    - Auto-pick language + voice based on content metadata
2. Agent K: Feedback & Analytics Collector
    - For each simulated post, generate dummy engagement stats:
        - Views, Likes, Shares, Comments (randomized but realistic)
    - Store metrics in analytics_db.json and link to original content ID
    - Create a feedback signal: "High-performing topics", "Underperforming formats"
3. Loop Hook: Adaptive Improvement Trigger
    - Create a placeholder function: adjust_future_content_strategy()
        - Reads top 3 performers of the week
        - Suggests content formats (e.g., more devotional tone on LinkedIn)
        - This will connect to future reinforcement learning in Task 5

File/Folder Additions

```
vaani-sentinel-x/
├── agents/
│   ├── publisher_sim.py          <-- expanded
│   ├── analytics_collector.py    <-- NEW
│   └── strategy_recommender.py   <-- NEW
```

```
├── analytics_db/
│   └── post_metrics.json          <-- NEW
```

Deliverables

- Updated repo with:
  - publisher_sim.py supporting 3 platform formats
  - analytics_collector.py generating and storing metrics
  - strategy_recommender.py suggesting improvements
- Sample JSON showing:
  - Posted content with engagement metadata
  - Feedback output from strategy agent
- Video or screenshots demonstrating the flow

Tech Stack Notes

- Platform Logic: Python
- Storage: JSON or basic MongoDB/Supabase (optional)
- Optional UI: Table showing posts + stats (React if time permits)