Adaptive Karma + Feedback-Driven Personality Simulation

Milestone Goal:

Make the financial agent simulation adaptive and evolving — integrating karma traits, behavior patterns, and feedback-based learning. The agents should grow in personality, modify strategies based on feedback, and produce long-term evolution in behavior.

Final Deliverables:

- Evolving karma scores across months.
- Personality labels (e.g., “The Frugal Monk”, “The Risky Warrior”) that change over time.
- Feedback system that allows:
  - Mentor feedback → modifies agent traits
  - User feedback → teaches the simulation to avoid repeating mistakes
- Improved output reports with karma evolution, behavior notes, and learning logs.

Step-by-Step Task Plan

Phase 1: Karma + Behavior Memory Expansion

1. Extend karma_tracker_agent
   - Add long-term memory: track total karma score over all months.
   - Define karma evolution thresholds:
     - Sattvic > 75 = “Wise Sage”
     - Mixed Rajasic/Sattvic = “Disciplined Hustler”
     - Mostly Tamasic = “Reckless Drifter”
   -
   - Save this evolution in data/persona_profile.json.
2. Extend behavior_tracker_agent
   - Track behavior streaks: e.g., “Consistent Saver (4 months)”.
   - Flag negative patterns: “Debt spiral”, “Goal fatigue”.

    - Save to data/behavior_memory.json.

Phase 2: Feedback System (Mentor + User-Driven)

3. Create feedback_engine.py
    - Accept feedback in two ways:
        - Mentor-generated (auto): e.g., “You invested 80% in crypto — too risky.”
        - User input (manual or simulated): e.g., “Avoid debt-heavy strategies next time.”
    - Parse and log feedback in data/feedback_logs.json.

4. Feedback influence logic
    - Karma/behavior agents adjust scoring based on repeated feedback patterns.
    - Apply small karmic shifts (+/-) over time if feedback is consistently positive or negative.
    - Use mentor’s tone and strategy to auto-trigger feedback (e.g., “Avoid repeating that mistake.”).

Phase 3: Simulation Pipeline Integration

5. Update simulate_timeline
    - Insert feedback_engine at the end of each month’s cycle.
    - Run karma and behavior agents after feedback is processed, so they evolve.
    - Allow agent personalities to adapt: slightly change savings rate, risk tolerance, etc. based on karma/persona evolution.

6. Persona Engine
    - Build a simple function that assigns and updates agent “personas” (titles/roles) based on:
        - Karma trend
        - Behavior pattern

        - Mentor feedback
    - Save in data/persona_history.json.

Extended Goal

- Add a “Simulation Reflection Report” that summarizes:
    - Monthly karma score
    - Feedback summary
    - New persona title
    - Learning outcomes

Documentation

- Update README.md or INSTRUCTIONS.md:
    - How to give feedback
    - How karma evolves
    - How agent personas are assigned
    - How to interpret reports

Excellent. That makes perfect sense.

Going forward here’s how we’ll now extend the Financial Guru simulator into its full version by integrating the agentic swarm for stock market, commodity exchange, and decentralized exchange (DEX) modules. This upgrade transforms the simulator from a karma + behavior engine into a multi-agent financial ecosystem — aligned with real-world market dynamics.

Financial Guru — Full Integration

Objective

To evolve the simulator into a multi-layered Gurukul subject that combines karma-finance simulation with agent-based trading simulation/ All managed under the central Financial Guru engine, and linked with the Gurukul App’s dashboard via API.

PART A: Agent Swarm Integration into Financial Guru with Karan

1. Stock & Commodity Market Agents

- Integrate the existing stock and commodity agents as callable modules.
- Each agent (e.g. stock_buyer_agent, commodity_analyst_agent, risk_evaluator_agent) should:
  - Analyze market conditions
  - Recommend buy/sell/hold positions
  - React to karma/behavioral influence (e.g., a Tamasic user may ignore risk signals)
- (Eventually buy and sell on users behalf)

Deliverable:

`data/market_simulation.json`
Contains monthly simulated trades, P&L, risk exposure, and agent commentary.

2. DEX Agents Integration

- Plug in DEX trading agents (using pseudo wallet + token data).
- Simulate swaps, yield strategies, and liquidity provision decisions.
- Ensure karma-based filters affect risk tolerance (e.g., Sattvic users avoid highly speculative DeFi assets).

Deliverable:

`data/dex_simulation.json`

PART B: Simulator Engine Upgrade

3. Simulate Timeline Updates

- Refactor simulate_timeline() to:
  - Trigger trading agents after basic behavior + karma computation
  - Append results from market & DEX agents
  - Update net financial score

4. Linking Karma to Market Strategy

- Build a bridge between karmic traits and market strategy:
  - Sattvic → Balanced long-term investor
  - Rajasic → Active trader, high turnover
  - Tamasic → Idle, or follows trend without logic

Ref Output:

`data/financial_wellness_score.json`

PART C: Gurukul Dashboard API & Subject Hook

5. Expose Unified API (Flask/FastAPI Layer)

- Create endpoints to serve:
  - /user/{id}/finance/guru-summary
  - /user/{id}/finance/market-activity
  - /user/{id}/finance/dex-trades
  - /user/{id}/finance/karma-history

6. Dashboard Components for Rishabh/Gandhar

- Define props + structure for:
  - Agent Trade Timeline
  - Karma-Driven Strategy View
  - Interactive Portfolio Summary

PART D: Final Testing & Docs

7. Testing

- Run 3-cycle simulation with karma and market agents.
- Confirm that:
  - Karma changes impact strategy
  - Outputs are logged and parsed

    - APIs return correct responses for frontend

8. Documentation

- Create FINANCIAL_GURU_README.md including:
    - Architecture
    - File structure
    - Karma-strategy mappings
    - How to run & test

Extended Task

- Add training feedback interface to fine-tune agent response.
- Plug-in Uniguru for local fine-tuning.

Financial Guru — Multi-Agent Market & Karma Integration

Assignee: Financial Simulator Developer

Collaborators: Karan/Rishabh/Madhuram/Gandhar (Gurukul Frontend & Dashboard)

Objective

Transform the existing karma + behavior simulator into a complete subject under the Gurukul App called Financial Guru, with real-time simulations of stock, commodity, and DEX markets via agentic swarms — all influenced by karmic traits.

Workstreams & Deliverables

Part A: Agent Swarm Integration (Stock, Commodity, DEX)

- Integrate stock_analyst_agent, commodity_trader_agent, dex_liquidity_agent, etc.
- Make each agent respect karmic behavioral influence:
    - Sattvic: safe, long-term logic
    - Rajasic: fast-paced, emotional trading
    - Tamasic: erratic or idle behavior
- Log detailed simulations:

    - data/market_simulation.json
    - data/dex_simulation.json

Part B: Simulator Engine Upgrade

- Modify simulate_timeline() to insert market + DEX agents after karma/behavior tracking
- Output unified results to:
    - data/financial_wellness_score.json
    - data/user_portfolio.json

Part C: API + Frontend Integration

- Build FastAPI or Flask endpoints:
    - /user/{id}/finance/guru-summary
    - /user/{id}/finance/market-activity
    - /user/{id}/finance/dex-trades
    - /user/{id}/finance/karma-history
-
- Coordinate with Rishabh/Gandhar:
    - Share prop structure and sample responses for React dashboard
    - Recommend 3 core components:
        - Karma-Driven Strategy View
        - Agent Trade Timeline
        - Portfolio Summary Panel

Part D: Testing + Documentation

- Simulate 3 months of financial journey with karma + trading agents
- Ensure all outputs reflect agent reasoning + karma logic
- Document:
    - FINANCIAL_GURU_README.md

    - Summary of agents, flows, and test instructions

Extended Task

- Add agent feedback loop for karma refinement
- Integrate with local Uniguru engine (test prompt replacement)

Instructions to Rishabh/Gandhar

Prepare new components under Subject: Financial Guru in the Gurukul Dashboard:

1. GuruKarmaStrategy.tsx — View karma trait + strategy impact
2. GuruMarketTimeline.tsx — Monthly trade activity (stock/commodity/DEX)
3. GuruPortfolioSummary.tsx — Show simulated P&L, asset spread, karma-score