**Sprint-4**

**Introduction**

In this Sprint, I have added Crew AI agents for investment decision and also added integration tests and validated and also added the backtesting code. The following sections contain the user stories I worked on with a detailed description of the tasks I worked on.

**User Stories**

I worked on the following User Stories:

[**RDNN: RDNN-Style Recurrent Reinforcement Learner #590**](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590)

**Conditions of Satisfiability:**

* Environment Correctness: Unit tests validate state transitions, actions, and rewards.
* Model Validity: RNN policy outputs have correct dimensions and numeric ranges.
* Training Stability: Reward curves show non-divergent, improving behavior.
* Latency: End-to-end inference (model load + forward pass + JSON) ≤ 1 s.
* Decision Consistency: Crew AI agent always returns a valid recommendation.

**Definition of Done:**

* Gym environment, policy network, and RL scripts implemented and unit-tested.
* Training loop configured, logs metrics, and saves best checkpoints.
* Inference API returns correct JSON under latency budget.
* Crew AI DecisionAgent defined and verified end-to-end.
* Integration tests and documentation completed.

**Tasks**

[RDNN.1 Implement Trading Environment #591](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/591)

[RDNN.1.1 Create gym.Env subclass with obs/action/reward logic (4 ph) #592](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/592)

[RDNN.1.2 Add transaction cost and slippage modeling (3 ph) #593](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/593)

[RDNN.1.3 Write unit tests for state transitions and reward calculations (2 ph) #594](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/594)

[RDNN.2 Build RNN Policy Network #617](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/617)

[RDNN.2.1 Define LSTM encoder module in PyTorch/TensorFlow (3 ph) #618](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/618)

[RDNN.2.2 Add DQN head and PPO policy/value heads (3 ph) #619](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/619)

[RDNN.2.3 Validate forward pass using dummy data (1 ph) #620](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/620)

[RDNN.3 Configure RL Algorithm #668](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/668)

[RDNN.3.1 Integrate SB3’s RecurrentPPO or custom DQN wrapper (4 ph) #669](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/669)

[RDNN.3.2 Set and document hyperparameters (lr, batch size, buffer size) (2 ph) #670](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/670)

[RDNN.3.3 Hook up logging (TensorBoard or W&B) (2 ph) #671](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/671)

[RDNN.4 Train #672](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/672)

[RDNN.4.1 Run training on historical price data (4 ph) #673](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/673)

[RDNN.4.2 Evaluate periodically on hold-out data and tune (4 ph) #674](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/674)

[RDNN.4.3 Save best model checkpoints with early stopping (1 ph) #675](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/675)

[RDNN.5 Develop Inference API #676](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/676)

[RDNN.5.1 Write InferenceAgent script to load policy checkpoint (3 ph) #677](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/677)

[RDNN.5.2 Given a new T-bar window, compute and format {action\_probs} JSON (2 ph) #678](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/678)

[RDNN.5.3 Benchmark latency and optimize code path (2 ph) #679](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/679)

[RDNN.6 Crew AI DecisionAgent #680](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/680)

[RDNN.6.1 Define Crew AI task prompt mapping action\_probs to BUY/SELL/HOLD (2 ph) #681](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/681)

[RDNN.6.2 Test DecisionAgent with edge-case JSON inputs (2 ph) #682](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/682)

[RDNN.7 Integration & Validation #683](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/683)

[RDNN.7.1 Write end-to-end integration test: train → inference → decision (3 ph) #684](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/684)

[RDNN.7.2 Confirm metrics, JSON schema, and error handling (2 ph) #685](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/685)

[RDNN.7.3 Document setup, usage examples, and agent prompts in README (2 ph) #686](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/686)

[RDNN.8 Backtesting & Evaluation #687](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/687)

[RDNN.8.1 Prepare historical test sets and data conversion (4 ph) #688](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/688)

[RDNN.8.2 Implement backtesting harness to simulate the full inference + decision pipeline over history (6 ph) #689](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/689)

[RDNN.8.3 Calculate performance metrics (e.g., accuracy, return, drawdown) and analysis (4 ph) #690](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/690)

[RDNN.8.4 Generate a backtesting report and visualizations (4 ph) #691](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/691)

**Tasks I Worked On**

[RDNN.6 Crew AI DecisionAgent #680](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/680)

I have added the Crew AI agents with a description, purpose etc and got the recommendation from the Crew AI agents. The task is estimated at 4 person hours but it took me 12 person hours to complete.

[RDNN.7 Integration & Validation #683](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/683)

I have written integration tests along with unit tests and validated the code. The task is estimated at 7 person hours but it took me 8 hours to complete.

[RDNN.8 Backtesting & Evaluation #687](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/687)

I have added the backtesting code and was able to successfully backtest the strategy and also able to generate the performance metrics. The task was estimated at 18 hours but it took me 22 hours to complete.

**Summary Table of Work**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| UserStory GitHub Issue ID | User Story | Story Points | Task GitHub Issue ID | Task | Task Hours | Status | Actual Hours |
| [RDNN](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) | [RDNN-Style Recurrent Reinforcement Learner](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) |  | [RDNN.6](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/680) | [Crew AI DecisionAgent #680](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/680) | 4 | Completed | 12 |
| [RDNN](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) | [RDNN-Style Recurrent Reinforcement Learner](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) |  | [RDNN.7](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/683) | [Integration & Validation #683](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/683) | 7 | Completed | 8 |
| [RDNN](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) | [RDNN-Style Recurrent Reinforcement Learner](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) |  | [RDNN.8](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/687) | [Backtesting & Evaluation #687](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/687) | 18 | Completed | 22 |

**Summary Table of Commits**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Commit Number | Commit Description (exactly as in github) | User Story | Task |
| August 7th, 2025 | 1a9af6882ebe661ce5c08dfe0b80e66e7195f8af | [Add Crew AI functionality](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/pull/781/commits/1a9af6882ebe661ce5c08dfe0b80e66e7195f8af) | [RDNN](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) | [RDNN.6](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/680) |
| August 7th, 2025 | e9f341e1a5269ca06b4e0ad9d3c7da3079a1905d | [RDNN Backtesting code](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/pull/781/commits/e9f341e1a5269ca06b4e0ad9d3c7da3079a1905d) | [RDNN](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/590) | [RDNN.7](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/683)  [RDNN.8](https://github.com/Rivier-Computer-Science/AI-Agent-Stock-Prediction/issues/687) |