

GenAI for ArbitrageGainer model feedback

The following are some suggestions from the ChatGPT functional programming tutor that we implemented:

- Workflow Definition and Structure
- Use of Domain Events

Suggestion on Workflow Definition and Structure

1. Consistently apply terminology.
 - SUGGESTION: In our “Emit more orders for partially fulfilled orders” workflow there was one event that was named inconsistently with other workflows that used a similar event.
 - ACTION: Rename the inconsistent event to use uniform terminology.
2. Clarify actions
 - SUGGESTION: In our “Emit buy and sell market orders” workflow we have pseudo code that captures conditional logic. ChatGPT suggested defining this logic more explicitly or breaking it up into separate workflows.
 - ACTION: I kept this conditional logic pseudocode. I found this suggestion unhelpful as even the example workflows shown in the class slides used conditional logic pseudo code to define the flow of logic.

Suggestions on Use of Domain Events

1. Add contextual details to events
 - SUGGESTION: In multiple workflows we have defined a type called CrudRes which consists of a Success or Failure. ChatGPT made a suggestion to give more contextual detail to the labels Success and Failure in this type.
 - ACTION: Since this is in the context of a crud operation to potentially a DB, I renamed these labels as PersistSuccessful and PersistFailed.

During our next round of the feedback loop I saw that our design was lacking some domain errors, so I asked for examples of potential domain errors to add.

ACTION: Our team added several relevant Domain errors that could arise during our application runtime.