Equivalence Class Testing

Equivalence classes define partitions of input values that are expected to behave similarly. Here's a breakdown of the equivalence classes considered in the test cases:

- ItemActionServiceImpl
    - getItemActions
        - Item ID: Valid (existing) vs. Non-existent
    - saveItemAction, replaceItemAction, deleteItemAction
        - Item ID: Valid (existing) vs. Non-existent
        - Item Action Quantity: Valid positive, Minimum (0), Maximum (Long.MAX_VALUE - close to Long.MAX_VALUE)
- ItemSummaryServiceImpl
    - getItemSummary
        - Item Action Dates: Valid (current day), Pre-dated, Future-dated (optional)
        - Number of Item Actions: Existing, No Actions
        - Item Action Quantity: Positive (insertions), Negative (sales), Zero
- ItemServiceImpl
    - saveItem
        - Item Code: Unique, Duplicate, Empty/Null
        - Cost, Price: Valid positive, Negative, Zero
        - Quantity: Valid positive/negative, Zero
    - getItem, replaceItem
        - Item ID: Existing vs. Non-existent

Decision Table Testing

Decision tables capture the combinations of input conditions and their expected outcomes. The test cases utilize decision tables for scenarios where multiple conditions influence the system's behavior. This includes:

- ItemSummaryServiceImpl
    - Determining how item actions with different dates (current, pre-dated, future) are considered for calculating the daily summary.

State Transition Testing

State transition testing focuses on how the system transitions between different states based on events and actions. The test cases cover:

- Normal Operation vs. Error State transitions for all services, testing how the system handles successful data processing and recovers from database errors.

Use Case Testing

Use case testing verifies the system's behavior against its intended functionalities. The test cases cover main success scenarios for:

- ItemActionServiceImpl
    - Retrieving item actions for an item
    - Adding, updating, and deleting item actions
- ItemSummaryServiceImpl
    - Retrieving daily item summary
- ItemServiceImpl
    - Retrieving all items
    - Adding a new item
    - Retrieving a specific item