

Advantages over the old implementation:

Extensibility

Our implementation is easily extensible for future consumables that might be added. Future consumable entities can just implement the 'Consumable' interface without requiring modifications to existing code, this adheres to the Open/Closed Principle (OCP).

Simplified Hierarchy

Removing the abstract 'ConsumableItem' class simplifies the class hierarchy. There's no need for an extra layer of abstraction, which makes the codebase cleaner and easier to understand. This aligns with the Single Responsibility Principle (SRP) by ensuring that each class has a clear and specific purpose.

Disadvantages:

Refactoring Effort

Implementing these changes required quite a bit of refactoring of the existing codebase to remove the abstract 'ConsumableItem' class and update the 'consume' logic in consuming actions.

Testing Complexity

Our implementation required additional testing to ensure that all implementations of the 'Consumable' interface worked correctly, which also took time.