Assignment: Optimizing Unit Test Suites for a Supermarket POS System

Imagine you are working on the early stages of a comprehensive, enterprise-level **point-of-sale (POS) system** designed for supermarkets. If successful, this system will undergo continuous development and frequent releases for decades. To support its long-term success and sustainability, your task is to **analyze and optimize the unit test suite and the system under test** to maximize their value for software engineers.

Requirements & Constraints

- A set of **Behavior-Driven Development (BDD) style acceptance tests** has been provided.
- These acceptance tests **must remain unmodified and continue to pass** throughout your work.
- You are free to **refactor** both the **unit test suite** and the **system under test** but **must not** modify, disable, or break any acceptance tests.

Steps to Complete the Assignment

1. **Understand the System Requirements**
   - Read the `Catalog.feature` and `ShoppingCart.feature` files in the **SupermarketReceipt.Specs** project.
   - These files define the system's behavior from an end-user perspective and serve as acceptance criteria.
2. **Analyze the Unit Test Suite and System Under Test**
   - Review the **unit tests** in the **SupermarketReceipt.Tests** project.
   - Examine the **system under test** in the **SupermarketReceipt** project.
   - Identify **problems** in both the unit tests and the system under test.
3. **Document Your Findings**
   - Classify each type of issue you find in the unit test suite and the system under test.
   - Provide a description of each issue type along with a proposed solution.
   - You do **not** need to list every specific occurrence of a problem, just classify and describe the solution.
4. **Refactor & Improve**
   - Apply your proposed solutions by refactoring the **unit tests** and the **system under test** accordingly.
   - Ensure that all acceptance tests in **SupermarketReceipt.Specs** remain intact and continue to pass.
5. **Final Validation**
   - Confirm that the entire **SupermarketReceipt solution compiles successfully**.
   - Ensure **all tests pass** across all projects in the solution.

Submission Requirements

- A **report** listing the identified problem types and their solutions.
- The **refactored code** for both the unit test suite and the system under test.

- Verification that all acceptance tests remain unchanged and continue to pass.

Evaluation Criteria

- **Thoroughness**: How well you identify and classify issues in the unit test suite and the system under test.
- **Code Quality**: The effectiveness and clarity of your refactoring.
- **Preservation of Behavior**: Ensuring the BDD acceptance tests remain intact and passing.
- **Completeness**: The entire solution should compile and all tests should pass.