Unit testing of online shopping cart system

The purpose of input domain modeling for the login and logout functionalities in the Online Shopping Cart system was to identify all meaningful inputs, categorize them into equivalence classes, and systematically test how the functions behave under various conditions.

**Input Domain modelling steps**

1. **Understanding Functionality**:  
   We began by analyzing the requirements and behavior of the **login** and **logout** functions:
   * The login function requires a username and password for authentication, and it can handle registration if the username is not found.
   * The logout function interacts with a cart object and seeks user confirmation to proceed.
2. **Defining Input Variables**:  
   The inputs were broken down into components for testing:
   * **username**: A string that could be valid, invalid, empty, or duplicate.
   * **password**: A string subject to validation criteria (e.g., length, special characters, uppercase letters).
   * **cart**: An object that can be empty or contain items.
   * **User decisions**: Inputs like 'y' (confirm) and 'n' (cancel) for logout and registration prompts.
3. **Equivalence Classes**:  
   We identified groups of inputs with similar behaviors, including:
   * Valid and invalid login credentials.
   * Empty or duplicate usernames.
   * Weak vs. strong passwords.
   * Cart states (empty or populated).
   * User decisions during logout or registration.
4. **Boundary Cases**:  
   Special attention was given to edge cases, such as:
   * Empty username or password.
   * Passwords just meeting or failing validation criteria.
   * Logging out from an empty cart vs. a cart with multiple items.
   * Quitting the login process.
5. **Creating Realistic Scenarios**:  
   To mimic real-world usage, test cases reflected common user behaviors, such as:
   * Mistyped login credentials.
   * Attempting to register an already existing username.
   * Cancelling a logout after seeing cart contents.