Woodcraft Haven Test Summary

Week 4

Jon Hayden – CSS/450 – Professor Iles

# Test Case 1 – Searching for wood materials

User should be able to open a search page to do an advanced search with filters. Results should include all matching materials or tools based on the filter used or give all results if filter is not used. Filters include the name of the item, the category of the item, and the cost of the item.

## Possible errors:

1. Items returned in the results don’t match the search term
2. Items returned in the results have a cost that does not fall in between the filtered price
3. Items returned in the results filtered by name gives results that don’t match the name
4. Items returned in the results filtered by category don’t fall in to the selected category

## Results of testing:

1. When searching results by search term, all items returned from search for correct – **PASSED**
2. When searching results by price range, some results returned were outside the required range – **FAILED**
3. When searching results based on name, all items returned based on name were correct – **PASSED**
4. When searching results based on category, all items returned were in the expected category – **PASSED**

## Reason for Failure:

There appears to be an issue in the if/else statement checking for the upper and lower boundary of some of the results being returned. This should be corrected in several locations because it uses this logic when searching by name, by category, and without filters. An even better fix would be to move this logic to its own function instead of repeating the logic multiple times. If it was its own function, I would only have to fix that one issue and all the other times that logic is needed, it will be fixed.

# Test Case 2 – Selling Handcrafted Products

The application should be able to list all types of hand crafted items for sale. These items should be able to get viewed, added, and removed easily from the user’s cart. These items should be kept in a database that can be updated as necessary and stores all items for sale. The database should have columns for name, description, category, cost, and availability.

## Possible errors:

1. All items for sale should be displayed
2. Items can be added and removed from user’s cart
3. The database can get updated to add more items
4. The database can get updated to change the price

## Results of testing:

1. All the items in the database (CSV file) are displayed as expected – **PASSED**
2. Buttons on the UI allows the user to add and remove items from the database – **PASSED**
3. The database can get updated to add more items…**PASSED** by editing CSV file. However, the application doesn’t not have an admin mode for adding additional items.
4. The database can have prices updated…**PASSED** by editing CSV file. However, the application doesn’t have an admin role for updating items.

## Reason for Failure:

As shown above, you can get PASSED results for all tests as long as you edit the database manually. There is no way to update the items in the database through the application. You would only want to allow this by creating user roles so only an administrator can update the items. There are lots of ways to add this functionality using something like Keycloak to manage user roles and login information. The next step in the development of this application would be to allow user management. And of course, the CSV file needs to be replaced by a full functioning database.

# Test Case 3 – Shopping Cart Management

The shopping cart page should list all items in the shopping cart along with the price and quantity. Users should be able to remove items and increase/decrease the quantity. The total cost including tax and shipping should be easily seen. Payment options should be provide with selections including card or PayPal.

## Possible errors:

1. Unable to hit checkout button
2. Shopping Cart total inaccurate
3. Missing payment options

## Results of testing:

1. User was able to hit checkout button and was notified of successful checkout – **PASSED**
2. Adding and removing items from the shopping cart correctly updated the cart total – **PASSED**
3. The user has no options for payment - **FAILED**

## Reason for Failure:

The biggest reason for any of the failures I’m finding is due to unimplemented features. There are no payment features added in this version of the application. The next version should have a payment feature for users so they can safely and accurately purchase all the items in their cart. Every purchase should also be tied to a user management feature so the user can get purchase and shipping updates via email or text.