Team 27 Space Traders Test Cases

Author: Maria

Test Case ID: M5.1

Description: *This use case tests the functionality of buying items on the market. This is a crucial aspect of the game since buying items is a major component of space trader. During this use case, we will test if money is deducted properly when buying and that cargo is effectively updated. A warning message will display if there is insufficient space and/or credits.*

Traceability:

- *Buying*:
  - If sufficient funds and cargo space:
    - Money decreases by the correct amount of the item
    - Ship inventory is updated with the correct item
    - Cargo space is decreased
  - If insufficient funds and/or cargo space:
    - The appropriate warning message is made preventing the player from completing the transaction

Actual Test Case:

Precondition: *User wishes to trade and selects “Trade”*

1. The player tries to buy something and has sufficient space and credits
2. The player tries to buy something and has insufficient space but enough credits
3. The player tries to buy something and has sufficient space but not enough credits
4. The player tries to buy something but has neither sufficient space nor sufficient credits

Postcondition: *The player’s inventory is updated with the item if successful and not updated if not successful.*

Expected Output:

1. The inventory is properly updated with the item as well as cargo space and credits depleted
2. An error message says that you do not have enough space is displayed
3. An error message says that you do not have enough credits is displayed

4. An error message of some sort is displayed

Actual Output:

1. When trying to buy the laser canon (118 credits, 4 space), the transaction was successful. Inventory now shows the laser canon, credits were depleted (1500-118 = 1382), and cargo space was depleted (10 - 4 = 6).
2. When trying to buy cardboard armor (4 space), an error message is displayed saying that the player does not enough space
3. When trying to buy a ship (1327) credits but the player only has 832 credits, an error message is displayed saying that the player doesn't have enough credits
4. When trying to buy Duel Laser (304 credits and 2 spaces) while having only 155 credits and 1 space left, an error message saying that the player doesn't have enough fuel credits

Pass/Fail Results: Pass test cases with no issues.

Fault Analysis: N/A

Author: Anshul

Test Case ID: M6.1

Description: Bandit Encounter

Traceability: A bandit encounters the player to rob credits (money) from the player. They will demand a certain amount of the player's credits before the player can continue travelling to the desired destination. The player then has 3 options:

-- pay the bandit's demand and continue to the desired destination. If the player cannot afford the bandit's demands, then the player must give the bandit all the items in their inventory. If the player has no items, the bandit will damage the ship's health. Then the player continues to the target destination.

-- try to flee back to the previous region. The success of fleeing is dependent on the player's Pilot skill **(higher Pilot level, higher chance of escape)**. If the player successfully flees back to the original region, they should still lose the fuel required to travel initially, but they keep all their credits & items and they are safe. If the player fails to flee, the bandit will take all their credits and damage the health value of the player's ship.

-- try to fight off the bandit. The success of defeating the bandit is dependent on the player's fighter skill **(higher fighter level, higher chance of winning)**. Successfully fighting off the bandit will allow the player to travel as intended to the desired destination without any new consequences. Additionally, success will grant the player some of the bandit's credits as a reward for winning the fight. Failing to fight off the bandit will cost the player all their credits and should damage the health of the player's ship.

Actual Test Case:

- **Precondition:** Player chooses to travel to new region.
- Bandit encounter randomly occurs based on difficulty
    - Option 1: Player picks “Pay” option
    - Option 2: Player picks “Flee” option
    - Option 3: Player picks “Fight’ option
- **Postcondition:** varies based on actions and consequences detailed above.

Expected Output: Provide the correct values and behaviors for the Space Trader application based on the sequence of actions that will be taken. These values must be clear and explicit enough to determine whether an application passes or fails this case.

- Option 1: Player picks “Pay” option
    - Player credits decremented, or all items lost if insufficient credits.
    - Player proceeds to new region.
- Option 2: Player picks “Flee” option
    - Player flees successfully based on Pilot level.
        - Player loses fuel to travel
        - Player proceeds to previous region
    - Player fails to flee
        - Player credits = 0, health reduced
        - Player proceeds to new region
- Option 3: Player picks “Fight’ option
    - Player wins battle based on Fighter level.
        - Player gains some credits
        - Player proceeds to new region
    - Player loses the battle
        - Player credits = 0, health reduced
        - Player proceeds to new region

Actual Output: Document the values and behaviors that your application displays on this test case.

- Option 1: Player picks “Pay” option
    - Player credits decremented, or all items lost if insufficient credits.
    - Player proceeds to new region.
- Option 2: Player picks “Flee” option
    - Player flees successfully based on Pilot level.
        - Player loses fuel to travel
        - Player proceeds to previous region
    - Player fails to flee
        - Player credits = 0, health reduced
        - Player proceeds to new region
- Option 3: Player picks “Fight’ option
    - Player wins battle based on Fighter level.
        - Player gains some credits
        - Player proceeds to new region
    - Player loses the battle

- Player credits = 0, health reduced
- Player proceeds to new region

Pass/Fail Results: Passed test case with no issues.

Fault Analysis: N/A

Author: Stephen Zacks

Test Case ID: M3.1

Description: This test case tests the functionality of beginning a game through the player creation phase.

Traceability:

- Difficulty selection adjusts the number of skill points for allocation as well as the number of credits the player begins with.
- Character skill allocation input is checked for validity and the player is correctly informed of any issues. If the player enters too many or too few skill points, they will be told to reenter valid values. If the player enters an invalid amount, such as -1 or a letter, the player will be told to reenter valid values.
- The name that the player enters is attributed to the created character. If no name is entered, a default name is given.

Actual Test Case:

- **Precondition:** The player begins the game
- The player selects the “Medium” difficulty
- The player enters an invalid skill point amount such as -1 and no name and tries to proceed.
- The player enters a skill point combination that totals 13 points and no name and tries to proceed.
- The player enters a skill point combination that totals 11 points and no name and tries to proceed.
- The player enters a skill point combination that totals 12 points and no name and tries to proceed.
- **Postcondition:** Character is created with the player’s specified attributes or default attributes as needed.

Expected Output:

- Switching the difficulty to Medium adjusts the number of skill points available to 12.
- The first three attempts to create the character will result in alerts that the player has entered the skill points incorrectly. Then the player will be allowed to reenter values.

- The fourth attempt will allow the player to create their character. The screen will display their character with the input values as well as the default player name and the correct number of credits for the Medium difficulty.

Actual Output:

- The skill points available was correctly adjusted when the difficulty was changed.
- The first three attempts to create the character resulted in the player being alerted to reenter valid values.
- The fourth attempt allowed the player to be created.
- The created character was displayed with the correct skill points and the default name along with the correct number of credits.

Pass/Fail Results:

- The application successfully completed the test case.

Fault Analysis: N/A

Author: John

Test Case ID: M4.1

Description: This test case tests the functionality of traveling between regions.

Traceability:

- Distance and fuel cost calculation is correct.
- The players region is changed when they travel.
- The player is unable to travel to the same region they are in or any region with a fuel cost greater than fuel remaining
- The player must confirm if they want to travel to a region.

Actual Test Case:

- **Precondition:** The player begins the game and has made a player and is on the main screen.
- The player travels to region they are already in.
- The player clicks the travel button on a region that is reachable but cancels the action.
- The player travels to a new region they can afford.
- The player travels to a region that costs more fuel than available.
- **Postcondition:** The player is in a different region from where they started and has lost the appropriate amount of fuel for their travels.

Expected Output:

- All regions' distances are correctly calculated using the Pythagorean theorem, the square root of $x^2 + y^2$ is equal to the distance. Where x is the current regions x value minus the selected regions x value, and y is the current regions y value minus the selected regions y value.
- All regions' fuel costs are correctly calculated by dividing the distance by 1+ the players pilot skill, using integer division, and then dividing that number by 5.
- Selecting the region that you are in and clicking travel will result in a pop up box letting you know you can't travel to a region you are already in.
- Clicking the travel button when the selected region's fuel cost to travel is less than the fuel you currently have will result in confirmation box asking to confirm the travel. The player by canceling the travel will stay put and lose no fuel.
- Clicking the travel button when the selected region's fuel cost to travel is less than the fuel you currently have will result in confirmation box asking to confirm

the travel. Confirming the travel, the player will travel to the selected region and lose the appropriate amount of fuel.

- Selecting and trying to travel to a region where the fuel cost to travel is more than the fuel you currently will result in a pop up box letting you know you don’t have enough fuel to travel to the selected region.

Actual Output:

- Each distance was correctly calculated and displayed.
- Each fuel cost for every region was calculated and displayed correctly.
- The player was unable to travel to the same region they were in and was told via pop up box.
- The player was prompted if they would like to travel to a region within their reach but canceled the “travel” and was not charged the fuel cost associated with the region.
- The player then traveled to a new region and lost the appropriate amount of fuel.
- The player was then unable to travel to a region with fuel cost higher than fuel remaining resulting in a pop up box prompting the player of this.

Pass/Fail Results:

- The application successfully passed the test case presented.

Fault Analysis: N/A