

# CNT 4714 – Project 1 – Spring 2026

**Title:** “Program Assignment 1: Event-Driven Programming”

**Points:** 100 points

**Due Date:** **Sunday February 1, 2026 by 11:59 pm (WebCourses time)**

**Objectives:** (1) To practice event-driven programming using Java-based GUIs. (2) To refresh your basic Java skills. (3) To simulate (albeit at a very high-level) an enterprise application.

**Description:** Develop a Java application that creates a standalone GUI application that simulates an e-store. We’ll call our store Nile Dot Com... we’re not quite as big as Amazon.com!<sup>1</sup> The application will allow the user to place in stock items into a shopping cart and once all items are included, total all costs (including tax), produce an invoice, and append a transaction log file.

Your program development must include the following steps:

1. Create a main GUI containing the following components:
  - a. An area that allows the user to input data into the application along with the descriptive text that describes each input area.
  - b. An area that shows the current contents of the shopping cart.
  - c. A total of six buttons as shown below with functionality as described below. The various input fields and buttons on the interface are only accessible at certain points during a user’s interaction with the e-store.
2. An input file named “inventory.csv”. This is a comma separated file which contains the data that will be read by the application when the user makes a selection. Each line in this file contains five entries: an item id (a string), a quoted string containing the description of the item, an in stock status (a string), the quantity on hand (an integer), and the unit price for one of the item (a double). A sample file is provided for you on WebCourses. Feel free to create your own input file for testing purposes or augment the provided input file.
3. An output file (append only) named “transactions.csv” must be created that uniquely identifies and logs each user transaction with the e-store. The unique transaction id will be generated as a permutation of the current date/time when the transaction occurred (see below). Note that this file must use a .csv extension and not a .txt extension. We’ll point out why this should be the case in class/discussions.

## Restrictions:

- Your source file shall begin with comments containing the following information:  
**/\* Name: <your name goes here>**  
**Course: CNT 4714 – Spring 2026**  
**Assignment title: Project 1 – An Event-driven Enterprise Simulation**  
**Date: Sunday February 1, 2026**  
**\*/**

---

<sup>1</sup> Source = <https://www.wonderslist.com/top-10-largest-rivers/>

- We'll assume that a shopping cart can hold at most 5 separate items. You don't need to worry about handling more than 5 separate items/transaction.
- We'll assume that our client's know the sequence of events that need to occur. This will reduce the amount of error detection/correction that you need to implement (see more below).
- Assume the tax rate is 6%. Discount rates as shown below on page 6 (in picture for item 4).

**Input Specification:** The file “inventory.csv” as described above (see example below as well).

**Output Specification:** Output is to appear in the specified components of the GUI and various message boxes that appear, plus the contents of the “transactions.csv” log file that will be generated.

**Deliverables:**

**Submit all of the following via WebCourses no later than 11:59pm Sunday February 1, 2026. Zip everything into a single folder.**

- (1) A working copy of your source code (all .java files) and any necessary supporting libraries.
- (2) Include a file that contains (**at least**) the following nine screen shots. All of these are taken from the perspective of your executing application (i.e., from your GUI). Clearly label each screenshot.
  - a. The initial GUI instance (like number 2 in Additional Information below).
  - b. The GUI after the user has entered an item number (ID) and quantity and clicked the “Search For Item #X” button (like number 4 below).
  - c. The GUI after the user has added an item from screenshot (b) above but before searching for the next item (like number 8 below).
  - d. The GUI after the user has searched for the next item after the one in (b) above was added to the cart, but before adding this new item to the cart (like number 9 below).
  - e. The message dialog box (the invoice) when the user clicks the “Check Out” button for an order containing multiple items (like number 13 below).
  - f. The GUI and the message dialog box when the user has entered an item number (ID) and quantity where the quantity is greater than the number in stock (like number 7 and 18 below).
  - g. The GUI and the message dialog box when the item the user has selected is out of stock (like numbers 5 and 15 below).
  - h. The GUI and the message dialog box when the item the user has selected is not contained in the inventory file (like 6 below).
  - i. A pair of screenshots (like 10 and 11 below) that show the last item in the cart being deleted and the interface being reset after the deletion.

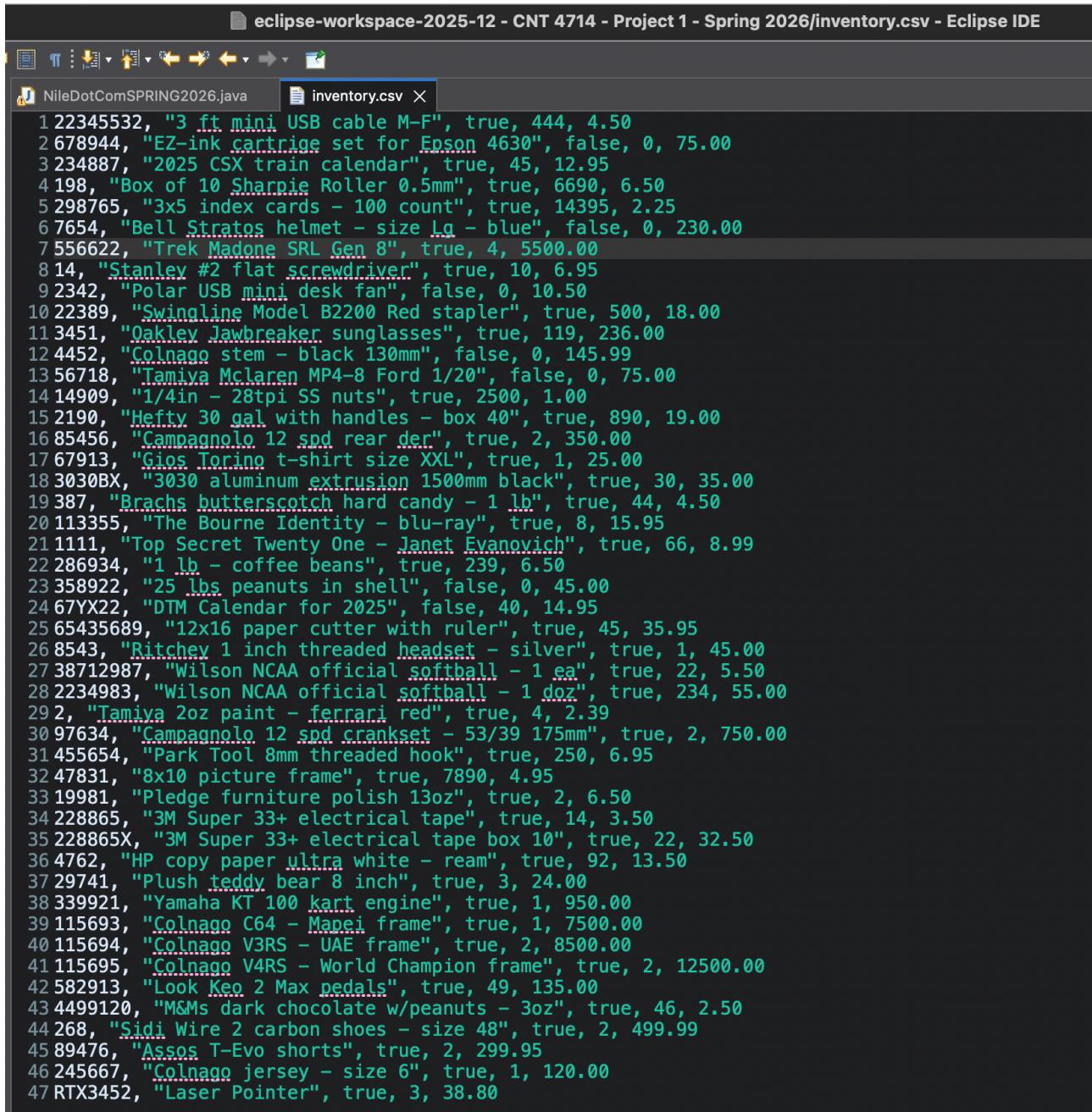
- You may include more screenshots than the nine listed above if you think they will illustrate additional features of your application than those already listed above.
- (3) Include your “transactions.csv” file as it stands after several different transactions have been recorded. You should have at least **5** different transactions in the file spanning a time period of at least **two hours** with at least **three** of these transactions having **four or five** items in the transaction. (similar to the one shown on the last page of this document).
- (4) Recommended configuration of your .zip folder is:

YourLastName\_YourFirstName\_CNT4714\_Project1.zip

- Source Code Folder
  - All .java files here + any supporting libraries needed
- Screenshot Folder
  - Screenshot a
  - Screenshot b
  - ...
  - Screenshot i (2 screenshots together – see (i.) above)
  - Additional screenshots you would like to provide – not required
- Your transaction.csv file meeting the requirements shown in (3) above.

## Additional Information:

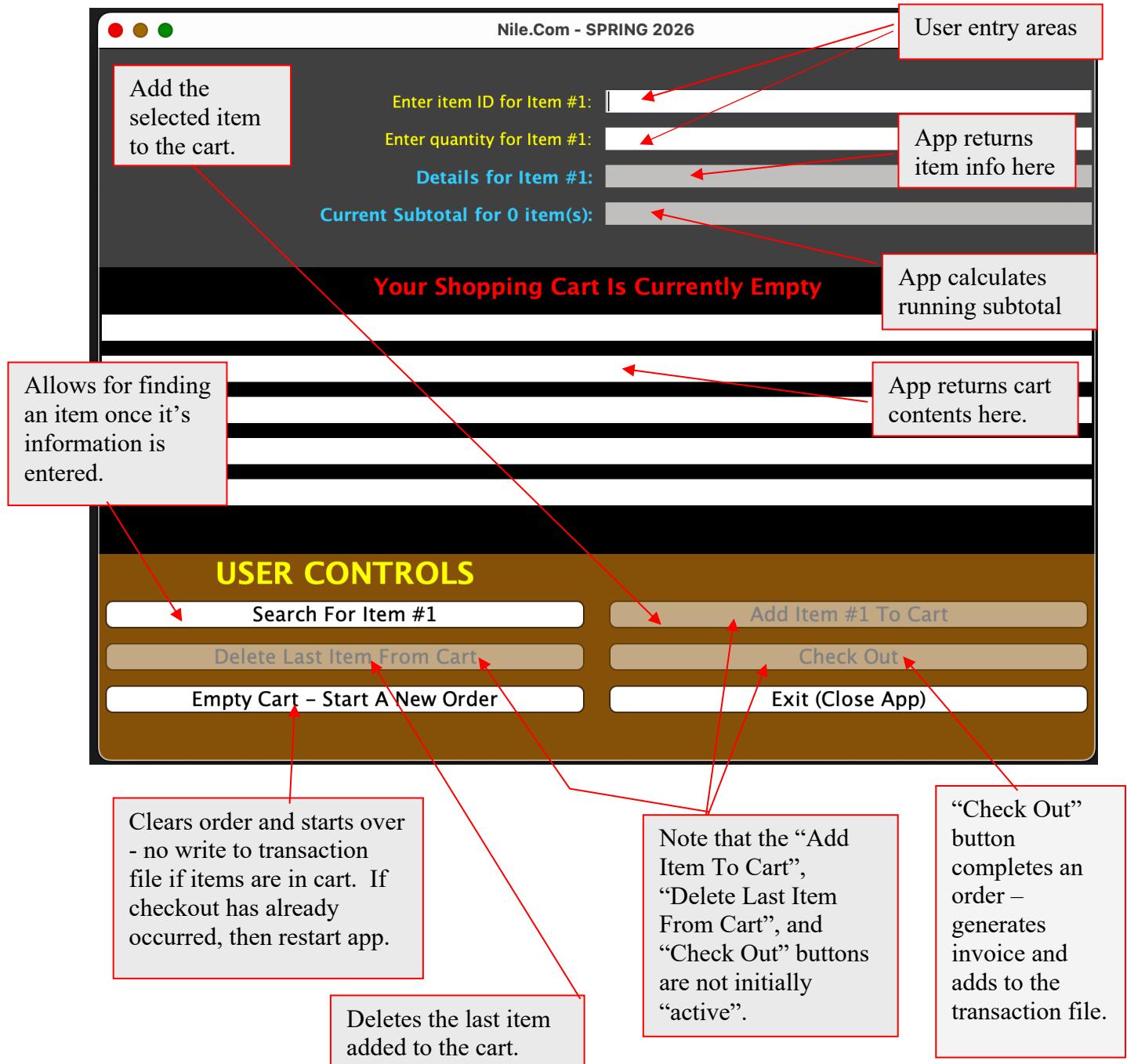
1. Screen shot of the contents of an example inventory.csv file.



The screenshot shows the Eclipse IDE interface with the title bar "eclipse-workspace-2025-12 - CNT 4714 - Project 1 - Spring 2026/inventory.csv - Eclipse IDE". Below the title bar, there is a toolbar with various icons. The main workspace contains two tabs: "NileDotComSPRING2026.java" and "inventory.csv". The "inventory.csv" tab is active and displays a list of 47 items, each with a line number, product ID, name, quantity, unit price, and total price. The data is as follows:

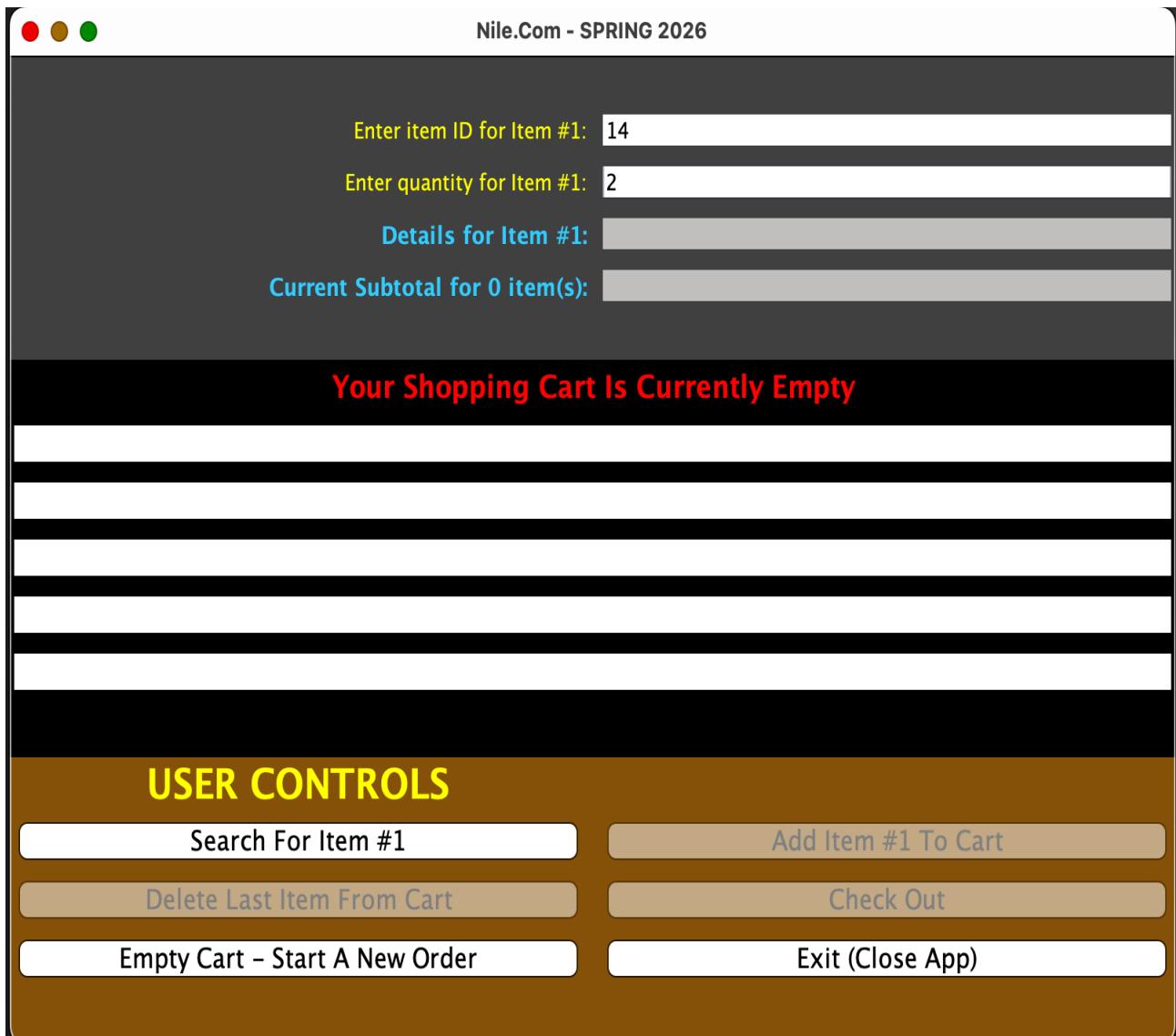
Line	ID	Description	Quantity	Unit Price	Total Price
1	22345532	"3 ft mini USB cable M-F"	true, 444	4.50	2000.00
2	678944	"EZ-ink cartridge set for Epson 4630"	false, 0	75.00	0.00
3	234887	"2025 CSX train calendar"	true, 45	12.95	582.25
4	198	"Box of 10 Sharpie Roller 0.5mm"	true, 6690	6.50	43795.00
5	298765	"3x5 index cards - 100 count"	true, 14395	2.25	32537.50
6	7654	"Bell Stratos helmet - size Lg - blue"	false, 0	230.00	0.00
7	556622	"Trek Madone SRL Gen 8"	true, 4	5500.00	22000.00
8	14	"Stanley #2 flat screwdriver"	true, 10	6.95	69.50
9	2342	"Polar USB mini desk fan"	false, 0	10.50	0.00
10	22389	"Swingline Model B2200 Red stapler"	true, 500	18.00	9000.00
11	3451	"Oakley Jawbreaker sunglasses"	true, 119	236.00	27404.00
12	4452	"Colnago stem - black 130mm"	false, 0	145.99	0.00
13	56718	"Tamiya McLaren MP4-8 Ford 1/20"	false, 0	75.00	0.00
14	14909	"1/4in - 28tpi SS nuts"	true, 2500	1.00	2500.00
15	2190	"Hefty 30 gal with handles - box 40"	true, 890	19.00	16810.00
16	85456	"Campagnolo 12 spd rear der"	true, 2	350.00	700.00
17	67913	"Gios Torino t-shirt size XXL"	true, 1	25.00	25.00
18	3030BX	"3030 aluminum extrusion 1500mm black"	true, 30	35.00	1050.00
19	387	"Brachs butterscotch hard candy - 1 lb"	true, 44	4.50	198.00
20	113355	"The Bourne Identity - blu-ray"	true, 8	15.95	127.60
21	1111	"Top Secret Twenty One - Janet Evanovich"	true, 66	8.99	589.34
22	286934	"1 lb - coffee beans"	true, 239	6.50	1558.50
23	358922	"25 lbs peanuts in shell"	false, 0	45.00	0.00
24	67YX22	"DTM Calendar for 2025"	false, 40	14.95	598.00
25	65435689	"12x16 paper cutter with ruler"	true, 45	35.95	1612.50
26	8543	"Ritchey 1 inch threaded headset - silver"	true, 1	45.00	45.00
27	38712987	"Wilson NCAA official softball - 1 ea"	true, 22	5.50	121.00
28	2234983	"Wilson NCAA official softball - 1 doz"	true, 234	55.00	12870.00
29	2	"Tamiya 2oz paint - ferrari red"	true, 4	2.39	9.56
30	97634	"Campagnolo 12 spd crankset - 53/39 175mm"	true, 2	750.00	1500.00
31	455654	"Park Tool 8mm threaded hook"	true, 250	6.95	1737.50
32	47831	"8x10 picture frame"	true, 7890	4.95	38515.50
33	19981	"Pledge furniture polish 13oz"	true, 2	6.50	13.00
34	228865	"3M Super 33+ electrical tape"	true, 14	3.50	49.00
35	228865X	"3M Super 33+ electrical tape box 10"	true, 22	32.50	715.00
36	4762	"HP copy paper ultra white - ream"	true, 92	13.50	1242.00
37	29741	"Plush teddy bear 8 inch"	true, 3	24.00	72.00
38	339921	"Yamaha KT 100 kart engine"	true, 1	950.00	950.00
39	115693	"Colnago C64 - Mapei frame"	true, 1	7500.00	7500.00
40	115694	"Colnago V3RS - UAE frame"	true, 2	8500.00	17000.00
41	115695	"Colnago V4RS - World Champion frame"	true, 2	12500.00	25000.00
42	582913	"Look Keo 2 Max pedals"	true, 49	135.00	6515.00
43	4499120	"M&Ms dark chocolate w/peanuts - 3oz"	true, 46	2.50	115.00
44	268	"Sidi Wire 2 carbon shoes - size 48"	true, 2	499.99	999.98
45	89476	"Assos T-Evo shorts"	true, 2	299.95	599.90
46	245667	"Colnago jersey - size 6"	true, 1	120.00	120.00
47	RTX3452	"Laser Pointer"	true, 3	38.80	116.40

## 2. Initial GUI:



Notice the current status of the various labels, input fields, and output areas in the initial GUI and notice how they change over the following pages as the user interacts with the application.

3. GUI after user has entered first item number and the quantity of the item they want to purchase, but before clicking the “Search For Item #X” button.



4. GUI after user has selected the first item and clicked the “Search For Item #1” button.

The screenshot shows a Mac OS X style window titled "Nile.Com - SPRING 2026". Inside the window, there are two input fields: "Enter item ID for Item #1:" containing "14" and "Enter quantity for Item #1:" containing "2". Below these is a text area labeled "Details for Item #1:" which displays "14 \"Stanley #2 flat screwdriver\" \$6.95 2 0% \$13.90". A red arrow points from the text "Information on the item selected/purchased has been extracted from the inventory file and written into the window on the line for item #1. Note proper currency formatting." to this details area. Another red arrow points from the text "Discount is based on the quantity of an item. If: 1-4 = 0% 5-9 = 10% 10-14 = 15% >=15 = 20%" to the same details area. At the bottom of the window, there is a section titled "USER CONTROLS" with five buttons: "Search For Item #1", "Add Item #1 To Cart", "Delete Last Item From Cart", "Check Out", and "Empty Cart - Start A New Order". The "Add Item #1 To Cart" button is highlighted with a white background and black text, while the others are in a standard greyed-out state. A red arrow points from the text "Note that the “Add Item #1 To Cart” button is now active and the “Search For Item #X” button is inactive. The “Delete Last Item From Cart” and “Check Out” buttons remain inactive." to the "Add Item #1 To Cart" button.

Nile.Com - SPRING 2026

Enter item ID for Item #1: 14

Enter quantity for Item #1: 2

Details for Item #1: 14 "Stanley #2 flat screwdriver" \$6.95 2 0% \$13.90

Current Subtotal for 0 item(s):

Information on the item selected/purchased has been extracted from the inventory file and written into the window on the line for item #1. Note proper currency formatting.

Discount is based on the quantity of an item. If:  
1-4 = 0%  
5-9 = 10%  
10-14 = 15%  
>=15 = 20%

**USER CONTROLS**

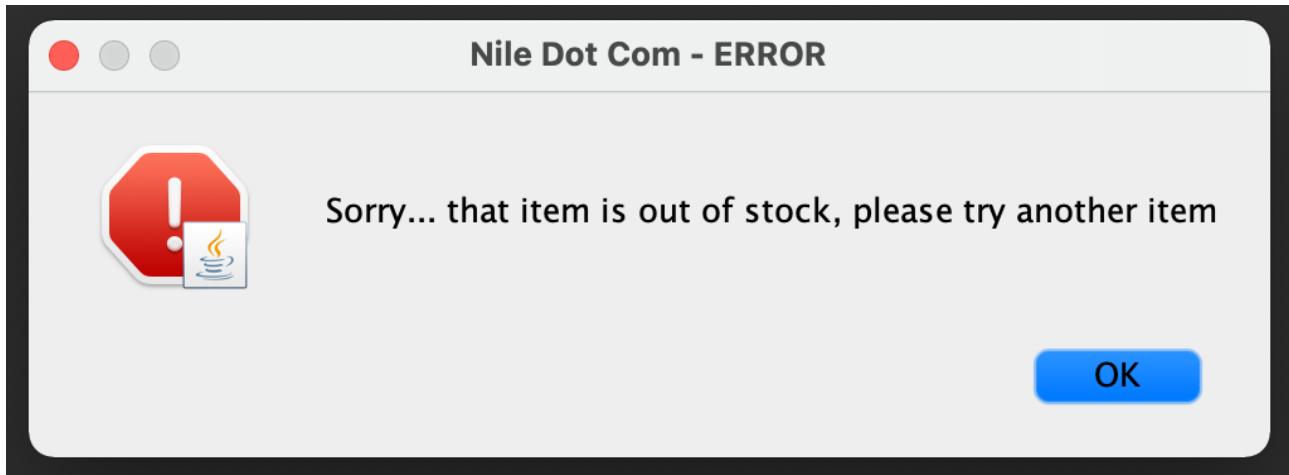
Search For Item #1      Add Item #1 To Cart

Delete Last Item From Cart      Check Out

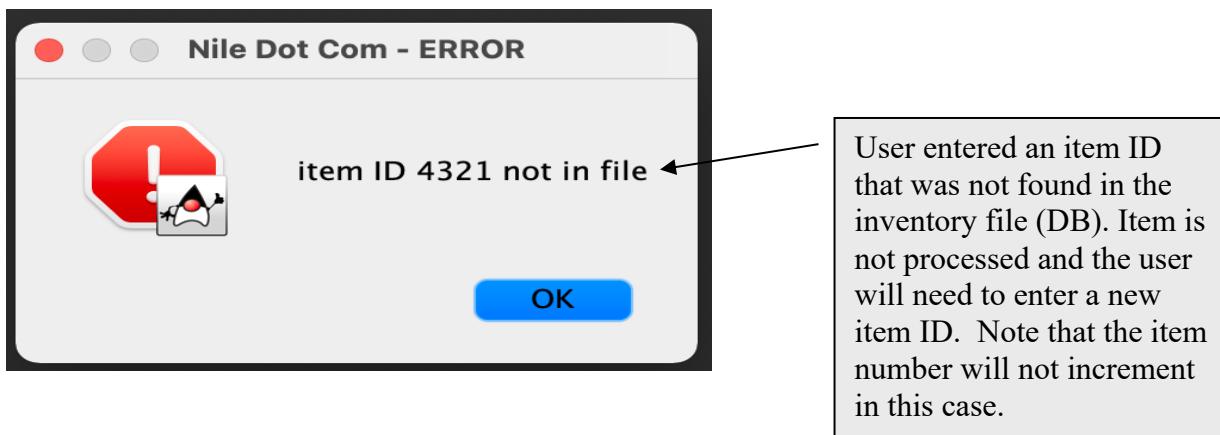
Empty Cart - Start A New Order      Exit (Close App)

Note that the “Add Item #1 To Cart” button is now active and the “Search For Item #X” button is inactive. The “Delete Last Item From Cart” and “Check Out” buttons remain inactive.

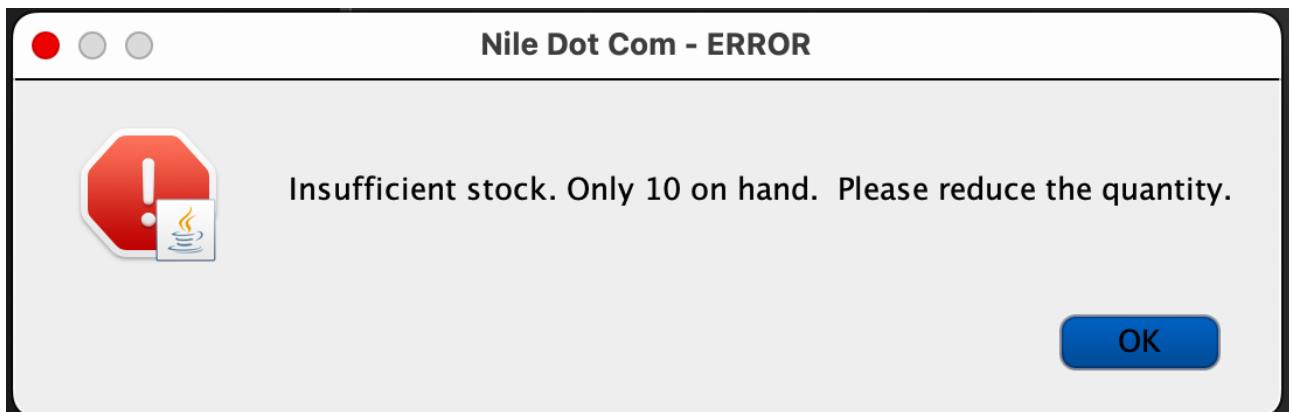
- When the user clicks on the “Search For Item #X” button and the item is found in the inventory, but the item is not in stock, then the following pop-up message would appear and the user is prompted to enter another product id. (see below as well)



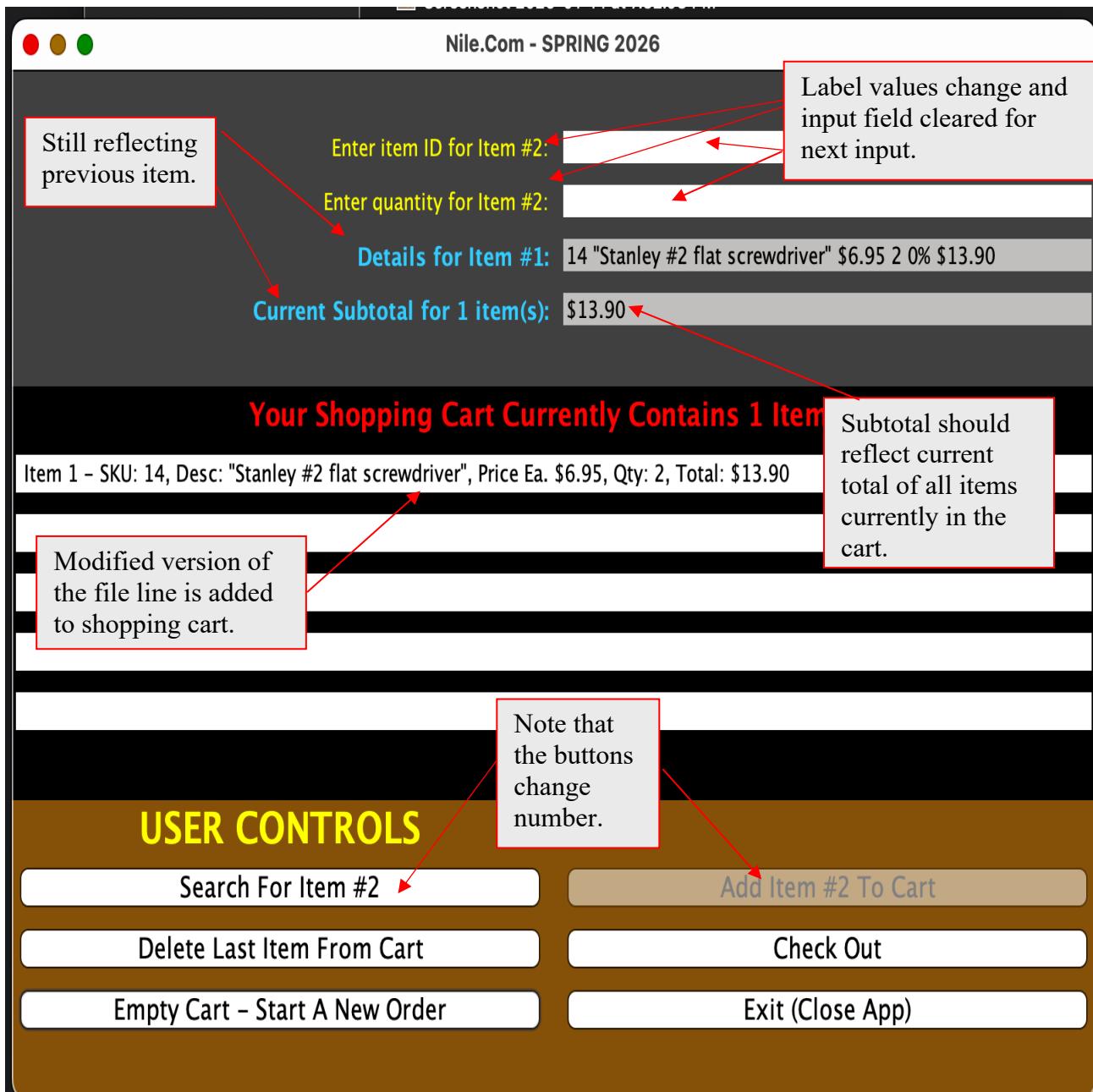
- When the user clicks on the “Search For Item #X” button, if the item is not found in the inventory, then the following pop-up message would appear on the user’s screen.



- When the user enters a quantity value larger than the quantity on hand, then the following pop-up message would appear on the user’s screen.



8. When the user clicks the “Add Item #1 To Cart” button, the GUI will be updated as shown:



9. User enters and finds (but has not yet added it to the cart) the next item in the order (which we'll make the last one for this order).

Nile.Com - SPRING 2026

Enter item ID for Item #2: 556622

Enter quantity for Item #2: 1

Details for Item #2: 556622 "Trek Madone SRL Gen 8" \$5,500.00 1 0% \$5,500.00

Current Subtotal for 1 item(s): \$13.90

Your Shopping Cart Currently Contains 1 Item(s)

Item 1 - SKU: 14, Desc: "Stanley #2 flat screwdriver", Price Ea. \$6.95, Qty: 2, Total: \$13.90

**USER CONTROLS**

Search For Item #2      Add Item #2 To Cart

Delete Last Item From Cart      Check Out

Empty Cart - Start A New Order      Exit (Close App)

10. The GUI after the user has entered the information for all the items and purchased the last item they wish to order.

The screenshot shows a window titled "Nile.Com - SPRING 2026". Inside, there are input fields for "Enter item ID for Item #3:" and "Enter quantity for Item #3:". Below these is a section labeled "Details for Item #2:" which displays "556622 "Trek Madone SRL Gen 8" \$5,500.00 1 0% \$5,500.00". A "Current Subtotal for 2 item(s):" is listed as "\$5,513.90". A bold red message "Your Shopping Cart Currently Contains 2 Item(s)" is centered. Below it, two items are listed: "Item 1 - SKU: 14, Desc: "Stanley #2 flat screwdriver", Price Ea. \$6.95, Qty: 2, Total: \$13.90" and "Item 2 - SKU: 556622, Desc: "Trek Madone SRL Gen 8", Price Ea. \$5,500.00, Qty: 1, Total: \$5,500.00". At the bottom, a "USER CONTROLS" section contains six buttons: "Search For Item #3", "Add Item #3 To Cart", "Delete Last Item From Cart", "Check Out", "Empty Cart – Start A New Order", and "Exit (Close App)".

Nile.Com - SPRING 2026

Enter item ID for Item #3:

Enter quantity for Item #3:

**Details for Item #2:** 556622 "Trek Madone SRL Gen 8" \$5,500.00 1 0% \$5,500.00

**Current Subtotal for 2 item(s):** \$5,513.90

**Your Shopping Cart Currently Contains 2 Item(s)**

Item 1 - SKU: 14, Desc: "Stanley #2 flat screwdriver", Price Ea. \$6.95, Qty: 2, Total: \$13.90

Item 2 - SKU: 556622, Desc: "Trek Madone SRL Gen 8", Price Ea. \$5,500.00, Qty: 1, Total: \$5,500.00

**USER CONTROLS**

Search For Item #3

Delete Last Item From Cart

Empty Cart – Start A New Order

11. When the user clicks the “Delete Last Item From Cart” button, the state of the GUI will change as shown below. Notice that Item #2 was removed from the shopping cart, the input area labels have been reset and the subtotal has been adjusted.

The screenshot shows a user interface for a shopping cart application. At the top, there are three colored dots (red, orange, green) followed by the text "Nile.Com - SPRING 2026". Below this is a form section with four input fields:

- Enter item ID for Item #2: [Input Field]
- Enter quantity for Item #2: [Input Field]
- Details for Item #2: [Input Field] (This field is circled in red)
- Current Subtotal for 1 item(s): \$13.90

Below the form, a bold red message states "Your Shopping Cart Currently Contains 1 Item(s)". Underneath this message, it lists "Item 1 - SKU: 14, Desc: "Stanley #2 flat screwdriver", Price Ea. \$6.95, Qty: 2, Total: \$13.90". A red oval highlights the empty space where Item #2 would normally be listed.

The bottom section is titled "USER CONTROLS" in yellow. It contains six buttons arranged in a grid:

Search For Item #2	Add Item #2 To Cart
Delete Last Item From Cart	Check Out
Empty Cart - Start A New Order	Exit (Close App)

Red ovals highlight the "Search For Item #2" button, the "Add Item #2 To Cart" button, and the "Delete Last Item From Cart" button.

12. Let's add a second item to our shopping cart before checking out.

Nile.Com - SPRING 2026

Enter item ID for Item #3:

Enter quantity for Item #3:

**Details for Item #2:** 85456 "Campagnolo 12 spd rear der" \$350.00 1 0% \$350.00

**Current Subtotal for 2 item(s):** \$363.90

**Your Shopping Cart Currently Contains 2 Item(s)**

Item 1 – SKU: 14, Desc: "Stanley #2 flat screwdriver", Price Ea. \$6.95, Qty: 2, Total: \$13.90

Item 2 – SKU: 85456, Desc: "Campagnolo 12 spd rear der", Price Ea. \$350.00, Qty: 1, Total: \$350.00

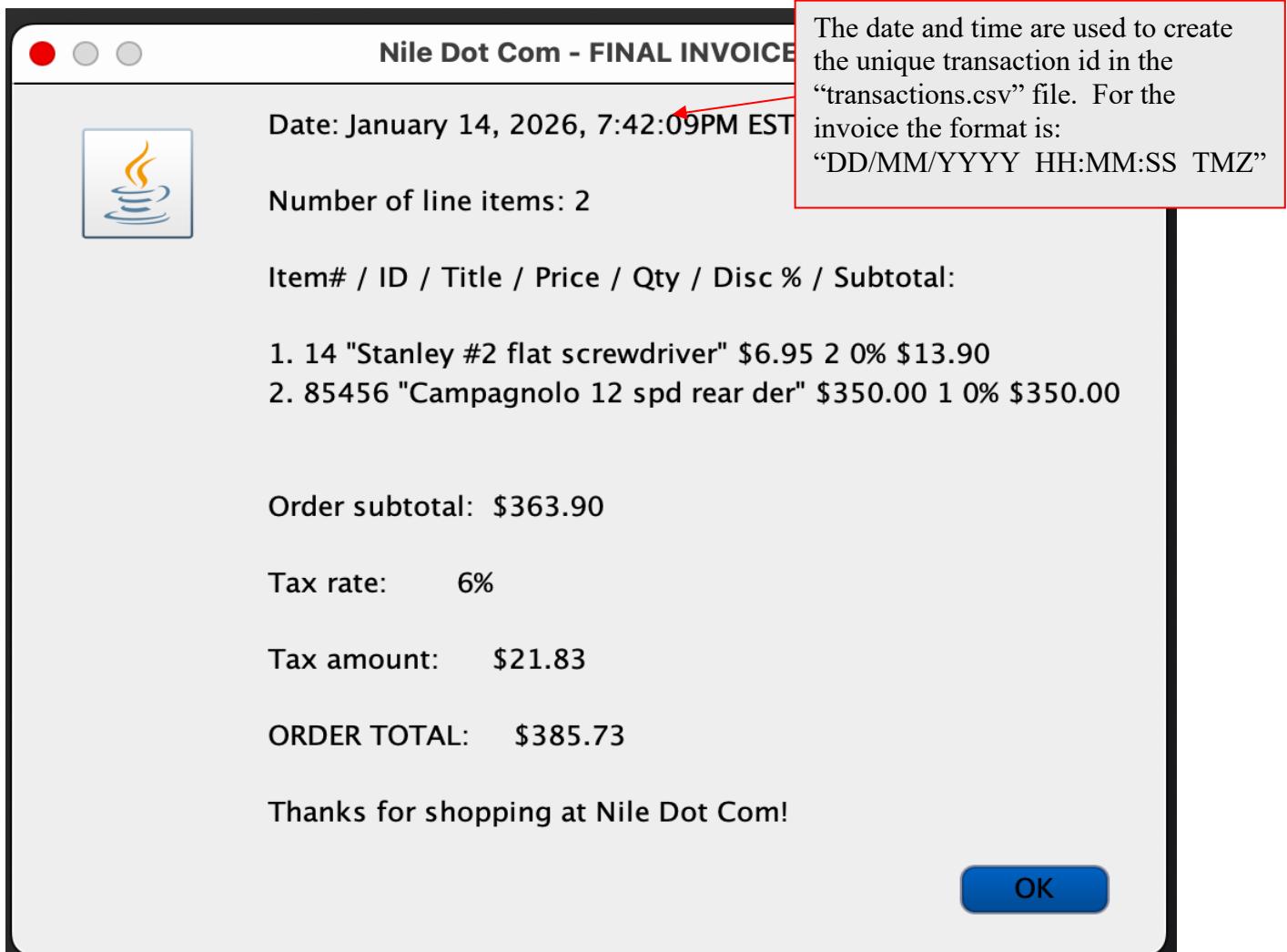
**USER CONTROLS**

**Search For Item #3** **Add Item #3 To Cart**

**Delete Last Item From Cart** **Check Out**

**Empty Cart – Start A New Order** **Exit (Close App)**

13. When the user clicks the “Check Out” button, the final invoice message should be generated and displayed. In addition, the order details are added to the transaction.csv file.



14. The transaction file after order shown above was finished. Note the unique transaction ids based on the date and time.

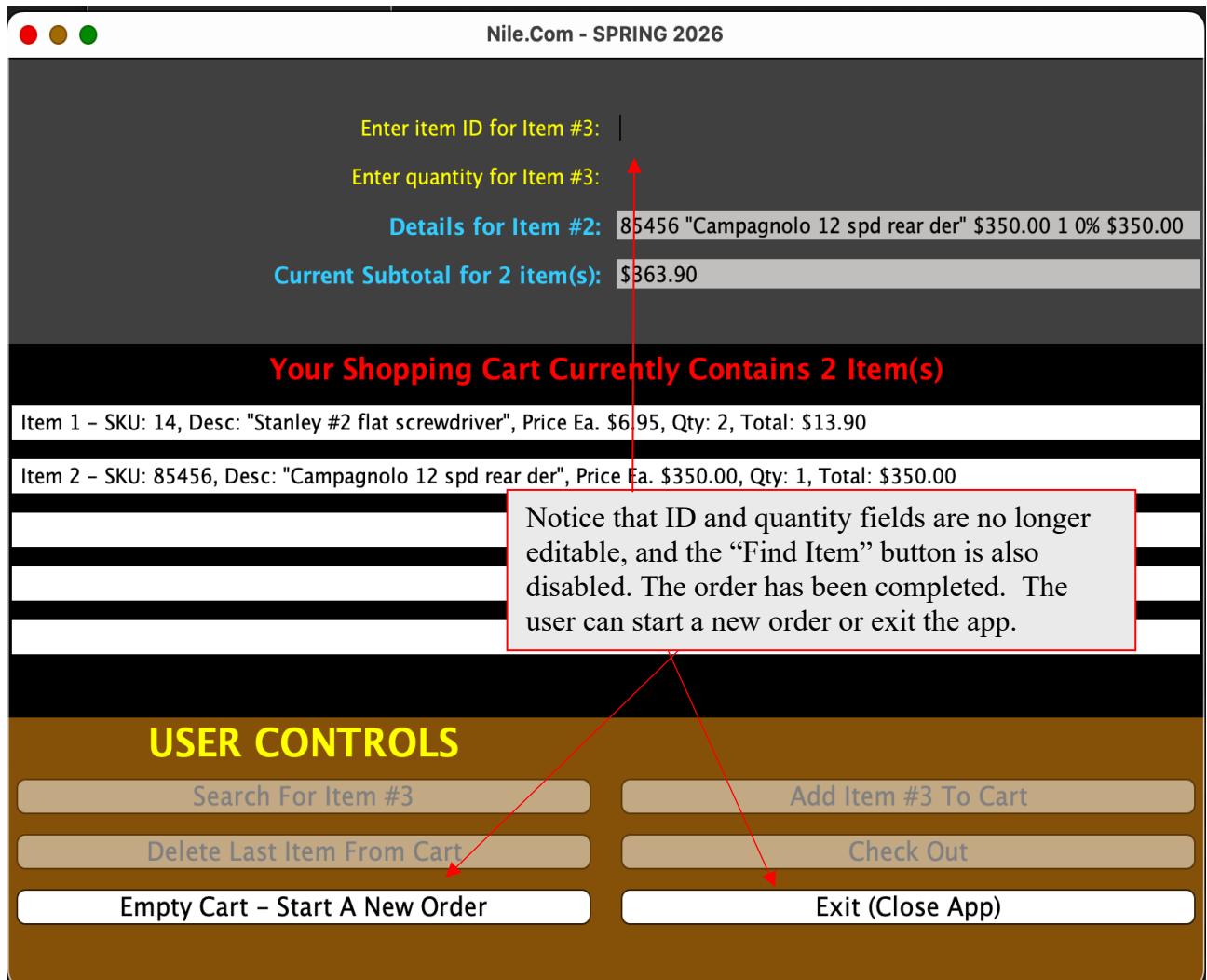
1	14012026194209	14	"Stanley #2 flat screwdriver"	6.95	2	0.0	\$13.90	January 14, 2026, 7:42:09 PM EST
2	14012026194209	85456	"campagnolo 12 spd rear der"	350.0	1	0.0	\$350.00	January 14, 2026, 7:42:09 PM EST
3								
4								

The transaction occurred on 14/01/26 at 19:42:09 (7:42:09 pm). See the invoice message above. Permutation used here is: DDMMYYYYHHMMSS.

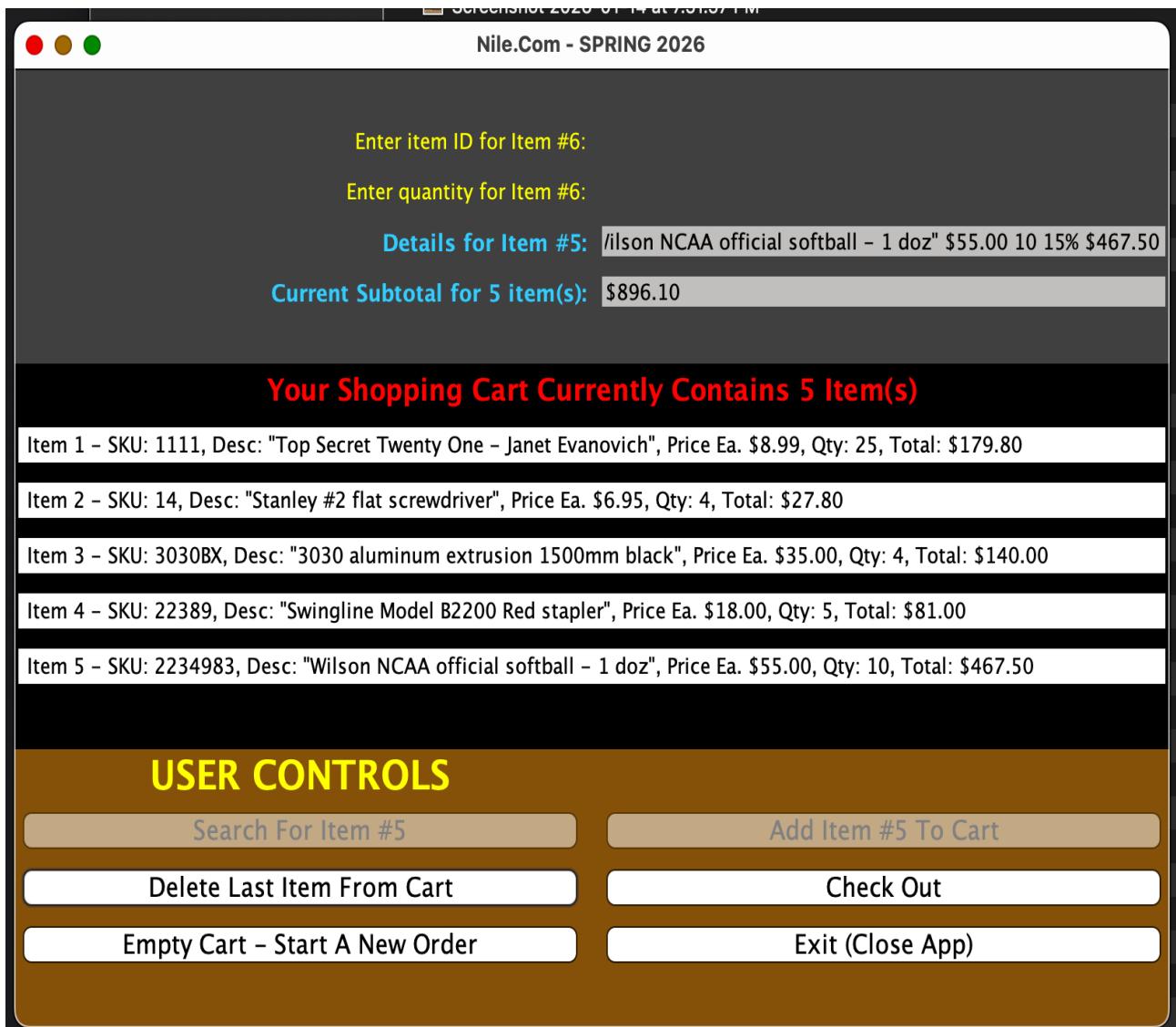
15. If the user enters an item number for an item that is not in stock (it is in the inventory – just not in stock – see number 5 above), the following message dialog should appear. Once the user clicks OK in the dialog box, then they would enter a new item number for another item and continue processing their current transaction. Notice below, that after clicking ok in the dialog box, the GUI is reset to the original item number and the input fields are empty.



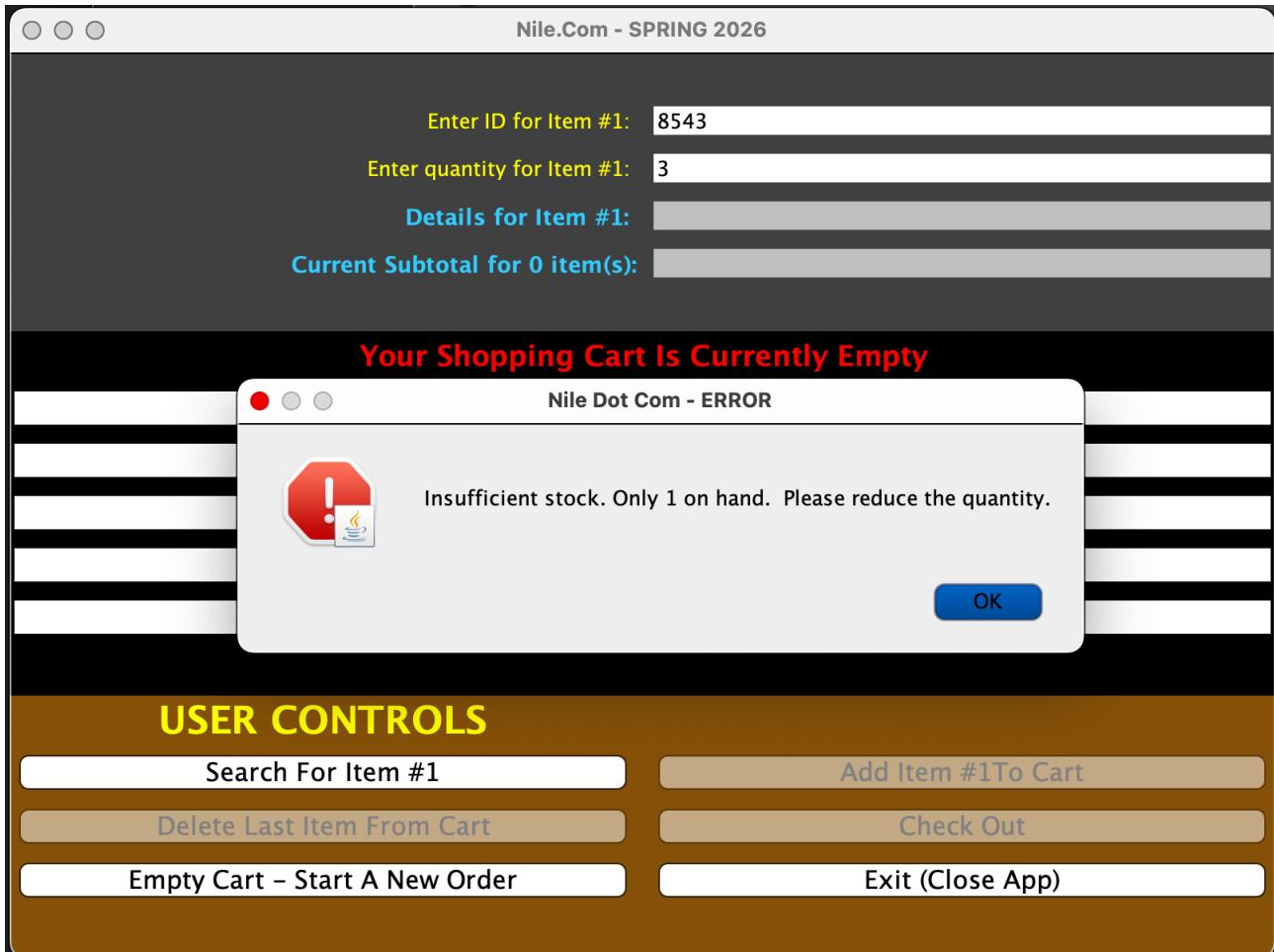
16. The GUI after the user has clicked the “Check Out” button and also clicked “OK” on the invoice dialog box.



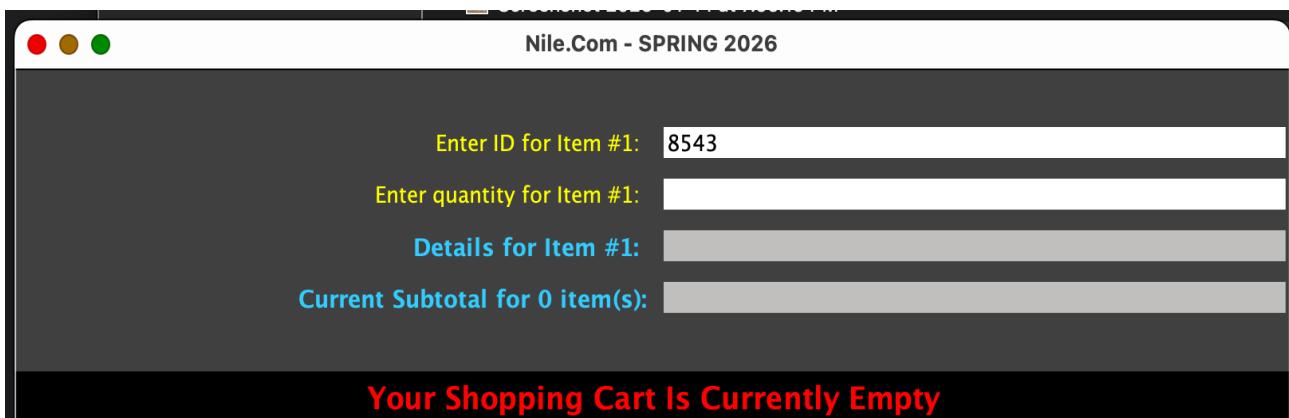
17. User has added 5 items to their shopping cart at this point. Cart is now full, so no more items can be added and the user must either delete the last item in the cart, check out or start a new order.



18. The situation may also arise where there is not a sufficient quantity in stock of an item that the user would like to add to the cart (see 7 above). When this situation arises, you should print the following message to the user.



In this case, once the user click the OK button on the dialog box, the GUI will clear the quantity field for a new quantity to be entered, but leave the item number intact (see cropped image below).



19. The transaction file after several orders have been completed. Note the unique transaction ids based on the date and time for every separate transaction. Note that the very first transaction in this file (from the first part of this document, does not include the item that we deleted from our cart, only the items that were in the cart at the time of the checkout. This file shows seven separate transactions over two different days.

```

eclipse-workspace-2025-12 - CNT 4714 - Project 1 - Spring 2026/transactions.csv - Eclipse IDE

File Edit View Insert Run Project Help
NileDotComSPRING2026.java inventory.csv transactions.csv X

1 14012026194209, 14, "Stanley #2 flat screwdriver", 6.95, 2, 0.0, $13.90, January 14, 2026, 7:42:09 PM EST
2 14012026194209, 85456, "Campagnolo 12 spd rear der", 350.0, 1, 0.0, $350.00, January 14, 2026, 7:42:09 PM EST
3
4 14012026195643, 1111, "Top Secret Twenty One - Janet Evanovich", 8.99, 25, 0.2, $179.80, January 14, 2026, 7:56:43 PM EST
5 14012026195643, 14, "Stanley #2 flat screwdriver", 6.95, 4, 0.0, $27.80, January 14, 2026, 7:56:43 PM EST
6 14012026195643, 3030BX, "3030 aluminum extrusion 1500mm black", 35.0, 4, 0.0, $140.00, January 14, 2026, 7:56:43 PM EST
7 14012026195643, 22389, "Swingline Model B2200 Red stapler", 18.0, 5, 0.1, $81.00, January 14, 2026, 7:56:43 PM EST
8 14012026195643, 2234983, "Wilson NCAA official softball - 1 doz", 55.0, 10, 0.15, $467.50, January 14, 2026, 7:56:43 PM EST
9
10 14012026195953, 8543, "Ritchey 1 inch threaded headset - silver", 45.0, 1, 0.0, $45.00, January 14, 2026, 7:59:53 PM EST
11
12 14012026221756, 582913, "Look Keo 2 Max pedals", 135.0, 2, 0.0, $270.00, January 14, 2026, 10:17:56 PM EST
13 14012026221756, 3451, "Oakley Jawbreaker sunglasses", 236.0, 1, 0.0, $236.00, January 14, 2026, 10:17:56 PM EST
14 14012026221756, 14909, "1/4in - 28tpi SS nuts", 1.0, 45, 0.2, $36.00, January 14, 2026, 10:17:56 PM EST
15
16 14012026221940, 556622, "Trek Madone SRL Gen 8", 5500.0, 1, 0.0, $5,500.00, January 14, 2026, 10:19:40 PM EST
17 14012026221940, RTX3452, "Laser Pointer", 38.8, 1, 0.0, $38.80, January 14, 2026, 10:19:40 PM EST
18 14012026221940, 298765, "3x5 index cards - 100 count", 2.25, 8, 0.1, $16.20, January 14, 2026, 10:19:40 PM EST
19 14012026221940, 1111, "Top Secret Twenty One - Janet Evanovich", 8.99, 1, 0.0, $8.99, January 14, 2026, 10:19:40 PM EST
20
21 15012026080537, 65435689, "12x16 paper cutter with ruler", 35.95, 1, 0.0, $35.95, January 15, 2026, 8:05:37 AM EST
22 15012026080537, 339921, "Yamaha KT 100 kart engine", 950.0, 1, 0.0, $950.00, January 15, 2026, 8:05:37 AM EST
23 15012026080537, 268, "Sidi Wire 2 carbon shoes - size 48", 499.99, 1, 0.0, $499.99, January 15, 2026, 8:05:37 AM EST
24
25 15012026080725, 286934, "1 lb - coffee beans", 6.5, 4, 0.0, $26.00, January 15, 2026, 8:07:25 AM EST
26 15012026080725, 47831, "8x10 picture frame", 4.95, 10, 0.15, $42.07, January 15, 2026, 8:07:25 AM EST
27 15012026080725, 198, "Box of 10 Sharpie Roller 0.5mm", 6.5, 1, 0.0, $6.50, January 15, 2026, 8:07:25 AM EST
28
29

```