Project: Self-Checkout Machine

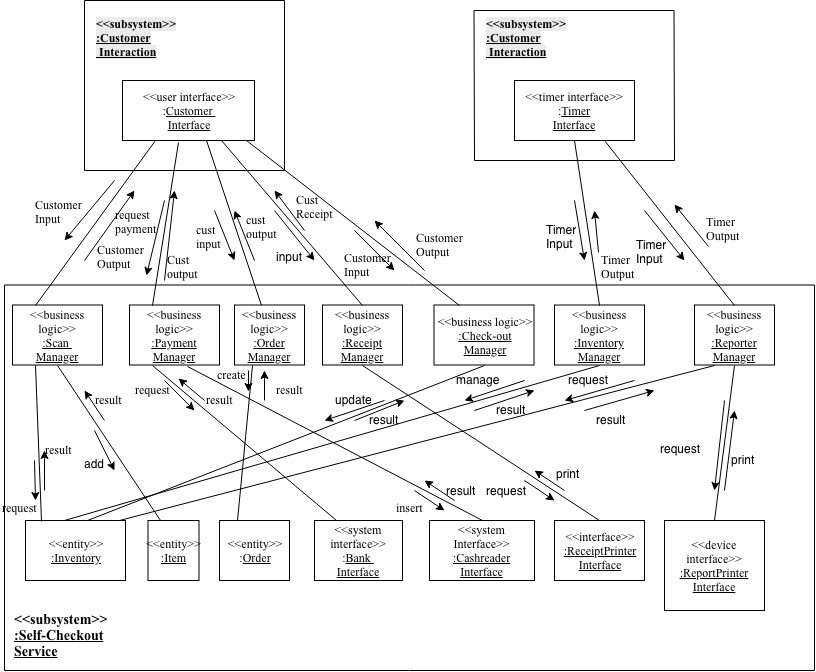
# 1.Project summary

A code that will go through all the processes that a self-checkout machine would do, such as check for the age in case the person is buying alcohol, remove items from inventory after purchase, and will allow for purchase through cash and credit cards. It also has a function for managers and staff to view and update shop inventory, such as adding new items or change price for an item. As well as printing out a daily report to see all the merchants they have sold daily.

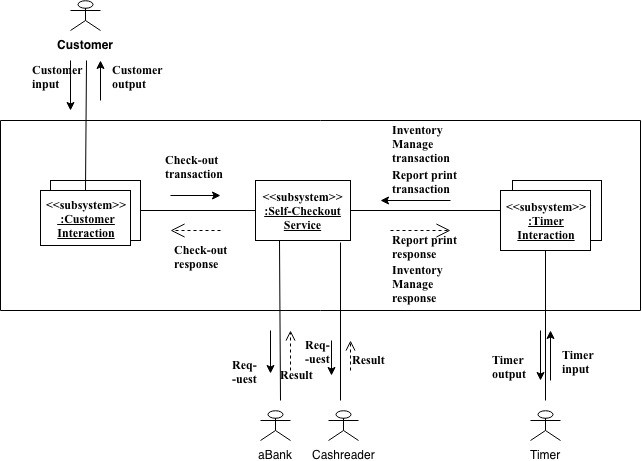
# 3. General Description

The Code will function as a database for a store and will keep track of inventory and prices of the items. It can then be used by employees and customers at self-checkout machines to purchase items, which will be updated in the inventory, and when new items arrive, they may be entered into the system. The System will allow for many types of payment such as cash and credit and will also provide the receipts. It will provide all basic functions a store would need.

# 4. Architecture of the project



The User interface takes in input from the user in order to interact with the classes such as the payment manager which in turn can interact with the entities in the system such as the items the user may be searching for. The classes can then give the desired output back out through the user interface. The classes may also interact with other interfaces such as printers and the bank system. Lastly there is a timer interface that will input the system to give out reports to the user interface or printer.



# 5. Use Case Diagram

# 

# 

# 

# 

# 

# 

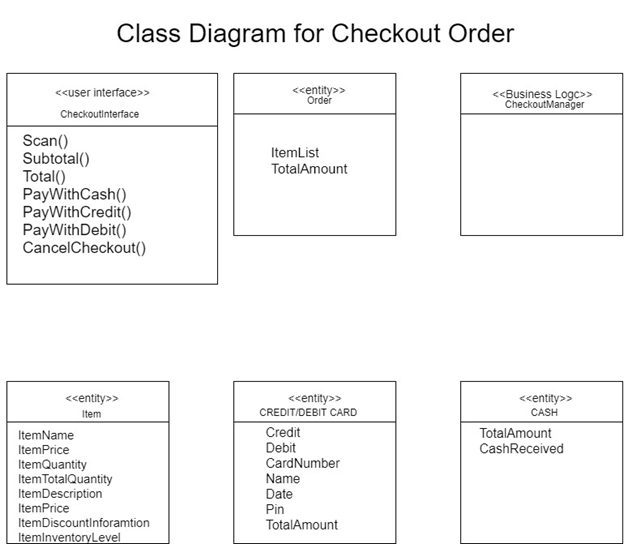
# 6. Use Case Specifications

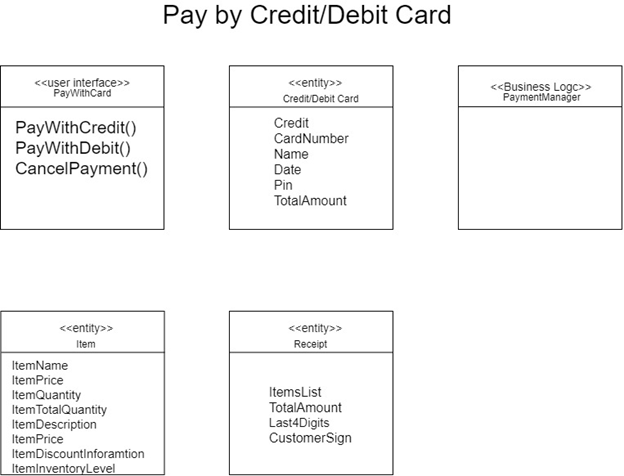
|  |  |
| --- | --- |
| Use Case Name: | Making payment |
| Actor(s): | Customer, System |
| Summary Description: | A customer can choose their payment  1)cash  2)credit card  3)debit card |
| Priority: | After scanning all the items, system will ask customers payment methods. |
| Status: | Active until the payment is complete. |
| Pre-Condition: | There must be an item in the shopping cart. |
| Post-Condition(s): | The payment has to to sufficient or authorized. |
| Basic Path: | 1. Scanning items done 2. Choose payment method 3. Showing total and balance 4. Payment done. |
| Alternative Paths: | 4b. The payment is not finished due to insufficient/card rejected/etc. |

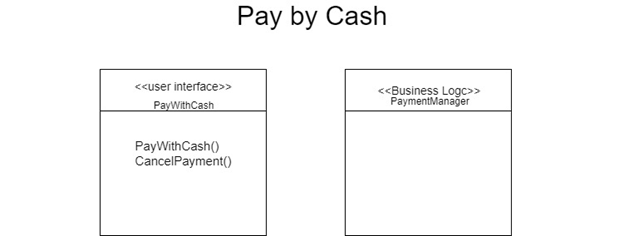
|  |  |
| --- | --- |
| Use Case Name: | ID |
| Actor(s): | Customer, System, Employee |
| Summary Description: | A customer goes through an ID check when trying to buy alcohol or  other age restricted items. |
| Priority: | Will begin before trying to pay for the items. |
| Status: | Active until the ID is accepted or rejected. |
| Pre-Condition: | There must be an item that has an age restriction in the customers  order. |
| Post-Condition(s): | The ID is either accepted and payment begins or rejected and the  item will be taken from the order. |
| Basic Path: | 1. An Item with an age restriction is in the customers order. 2. The customer is prompted to enter in their ID 3. The system will compare the age of the ID with the minimum age required.   4. The ID is accepted and the Customer is prompted for payment. |
| Alternative Paths: | 4b. The ID is rejected and the employee is alerted to remove the item. |

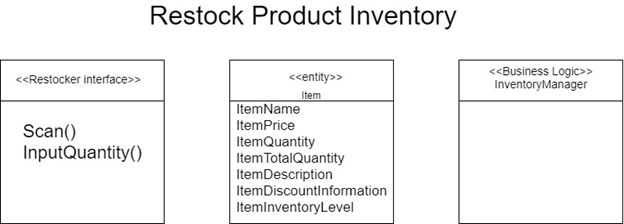
# 

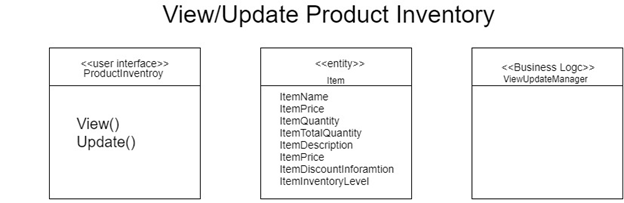
# 7. Class Diagram

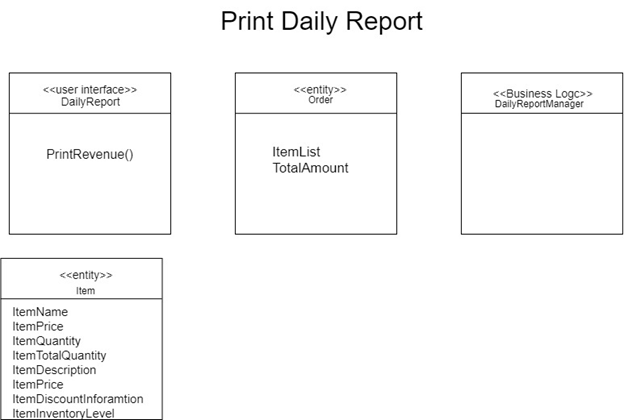


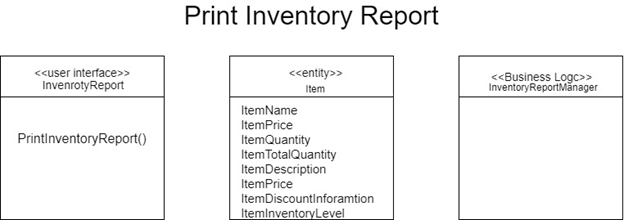






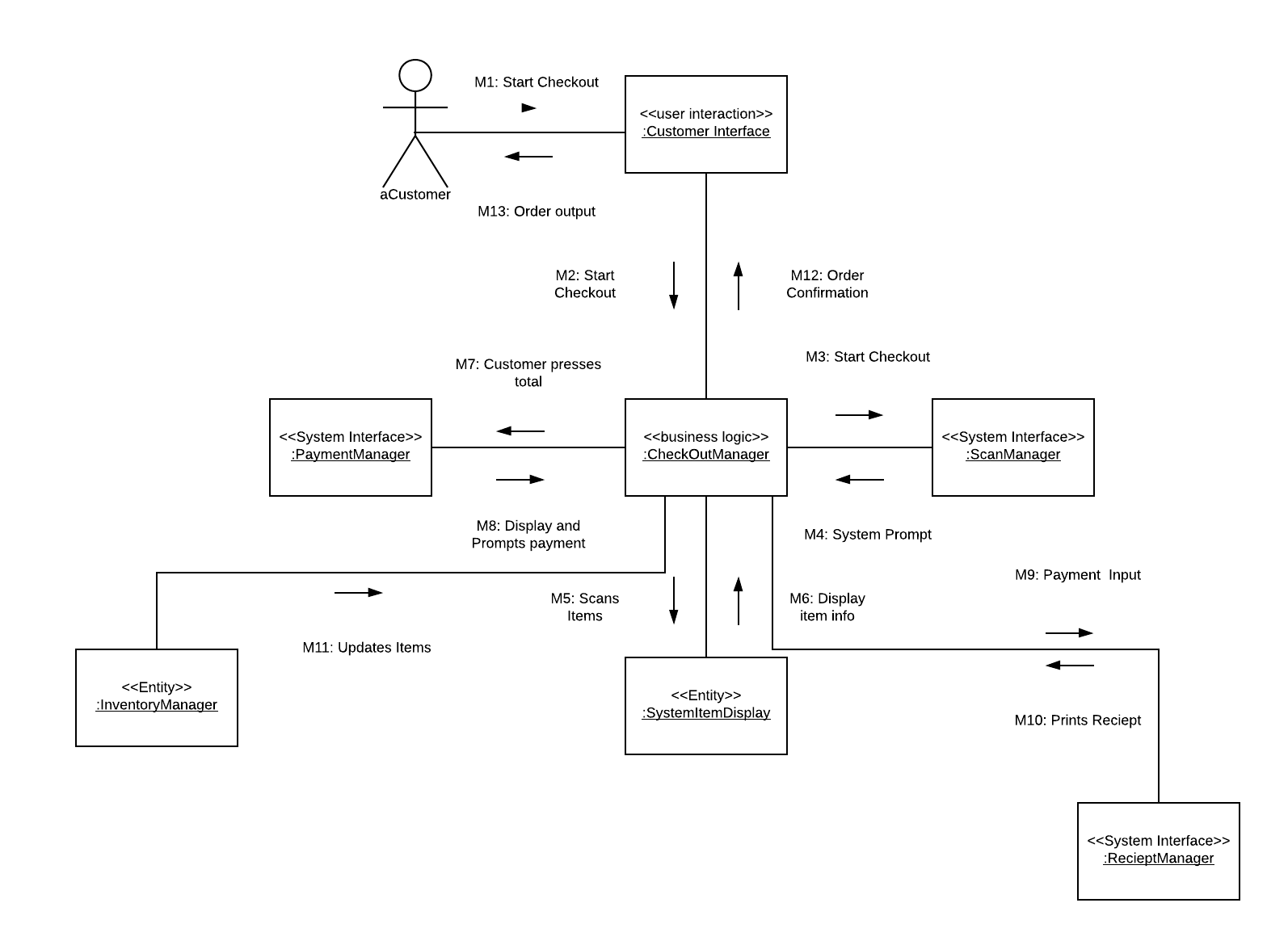




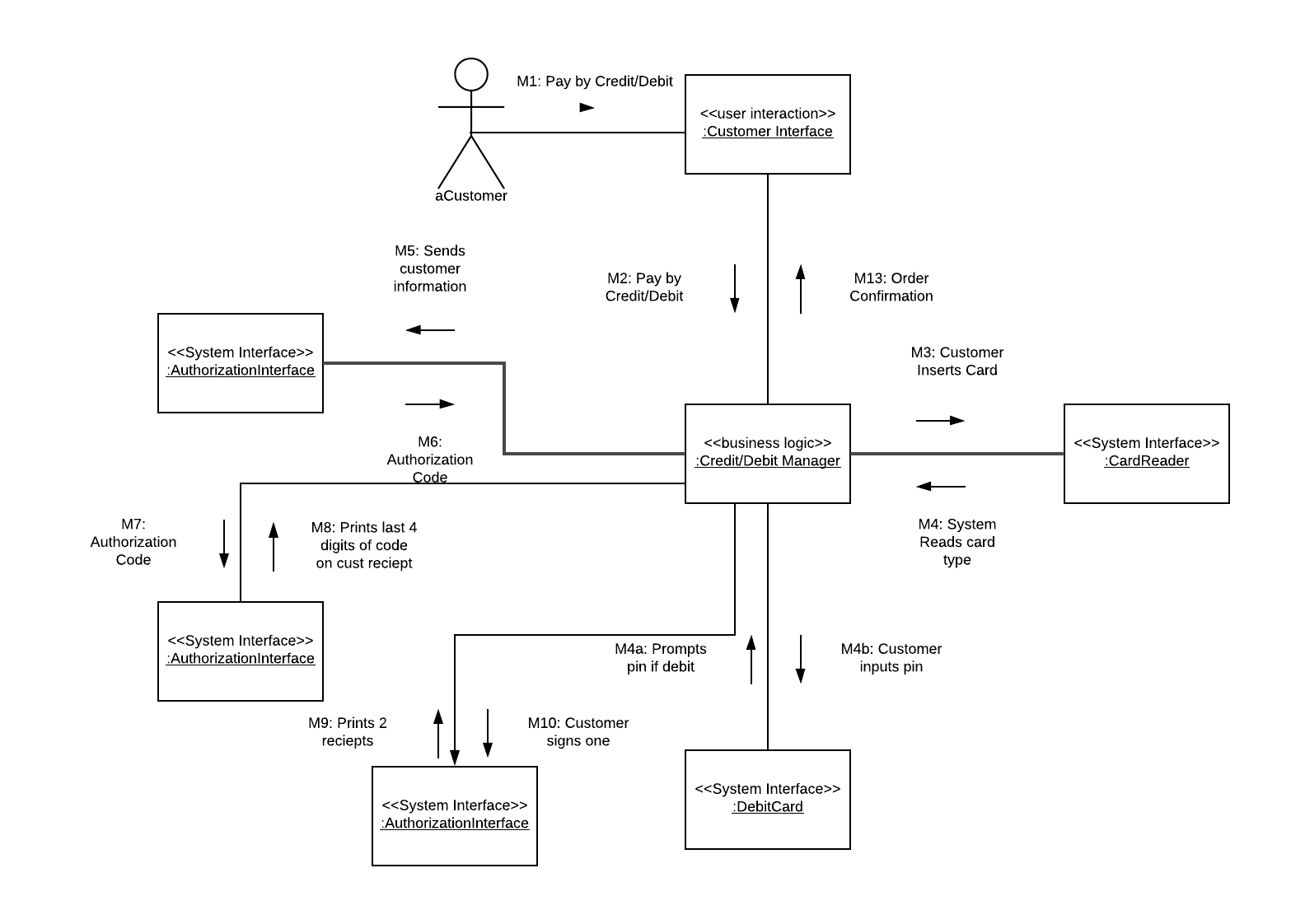


# 8. Sequence Diagrams

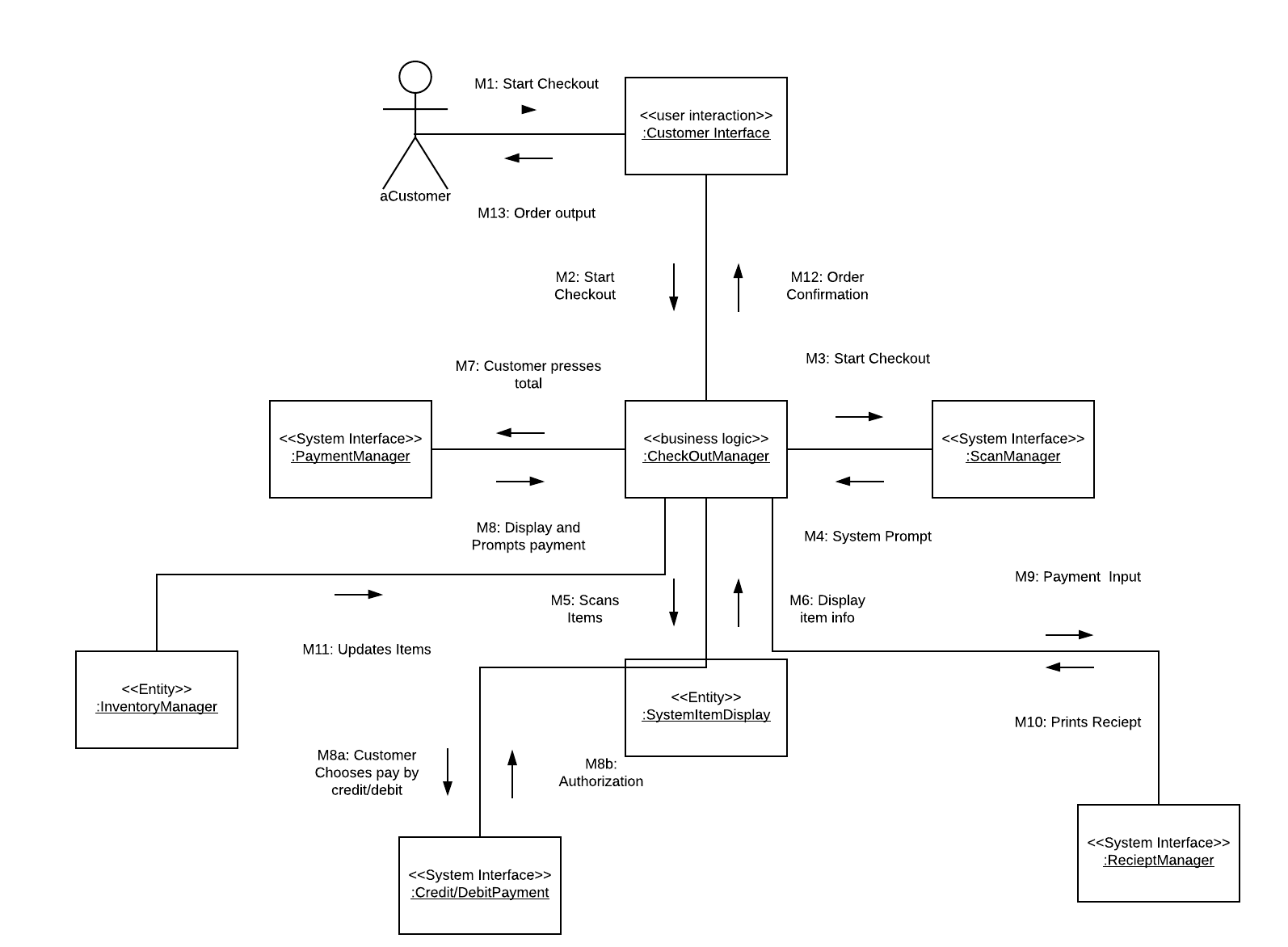
**Checkout Order Main Sequence**



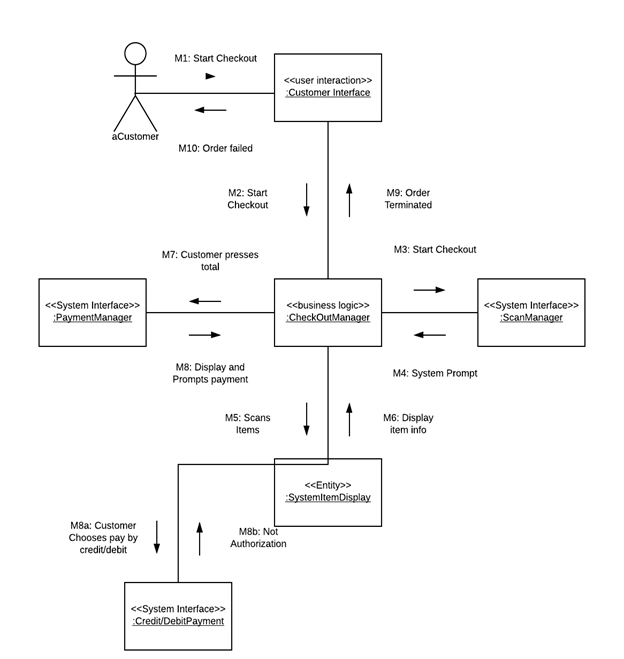
**Pay By Credit/Debit Main Sequence**



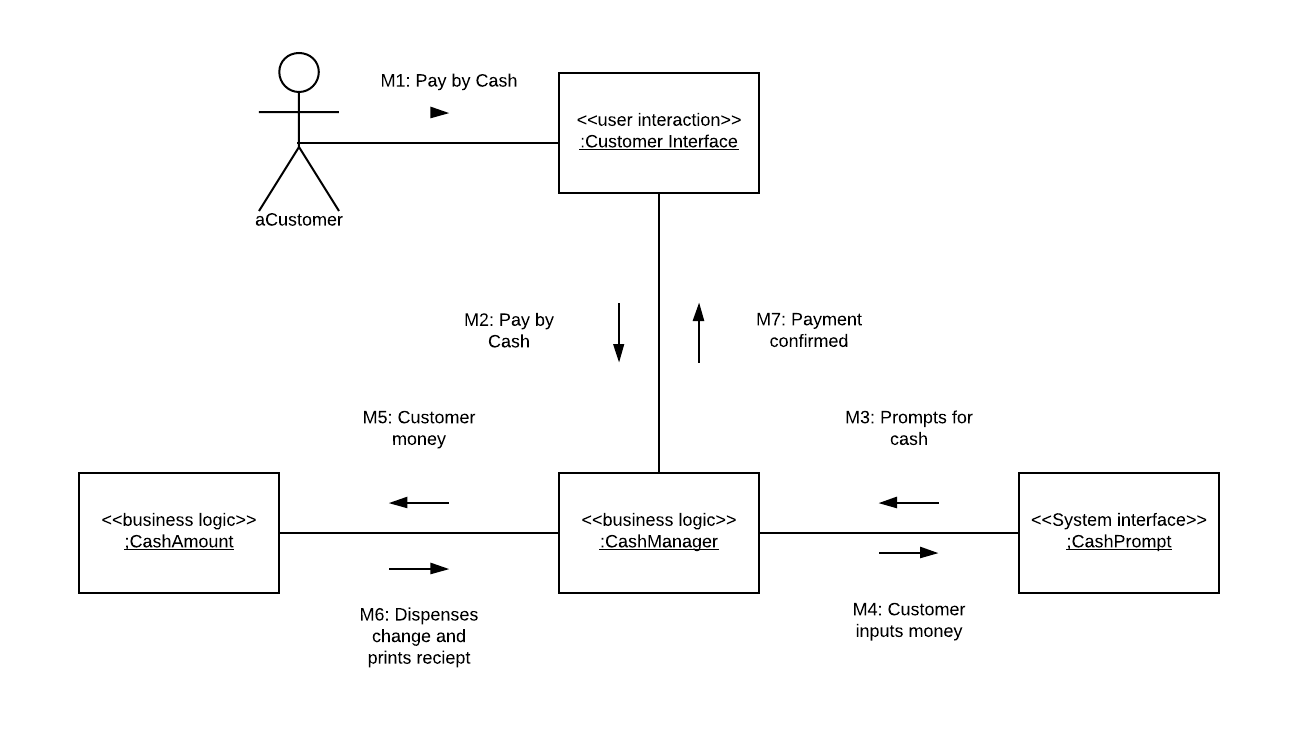
**Pay by Credit/Debit (Successful)**



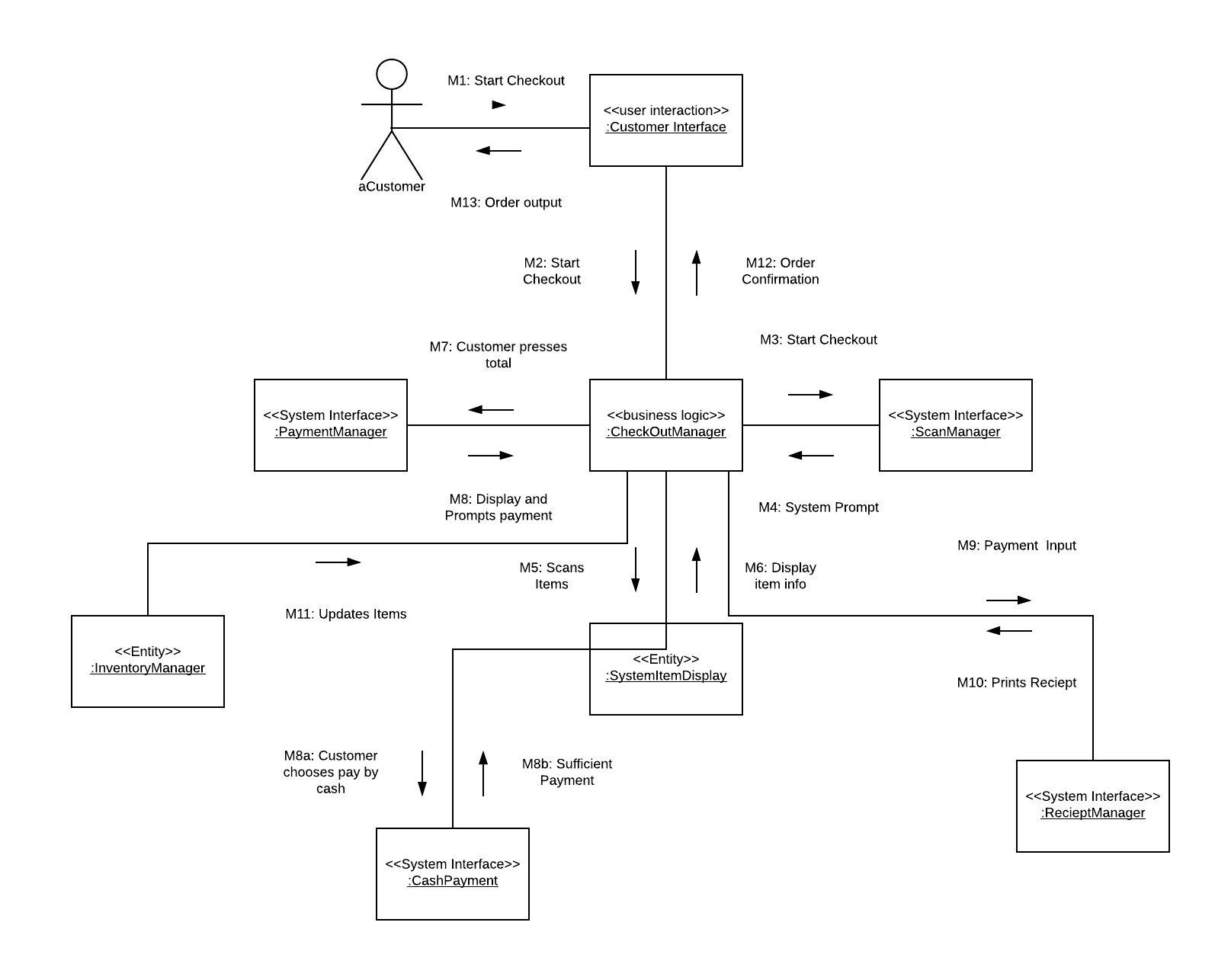
**Pay By Credit/Debit (Unsuccessful)**



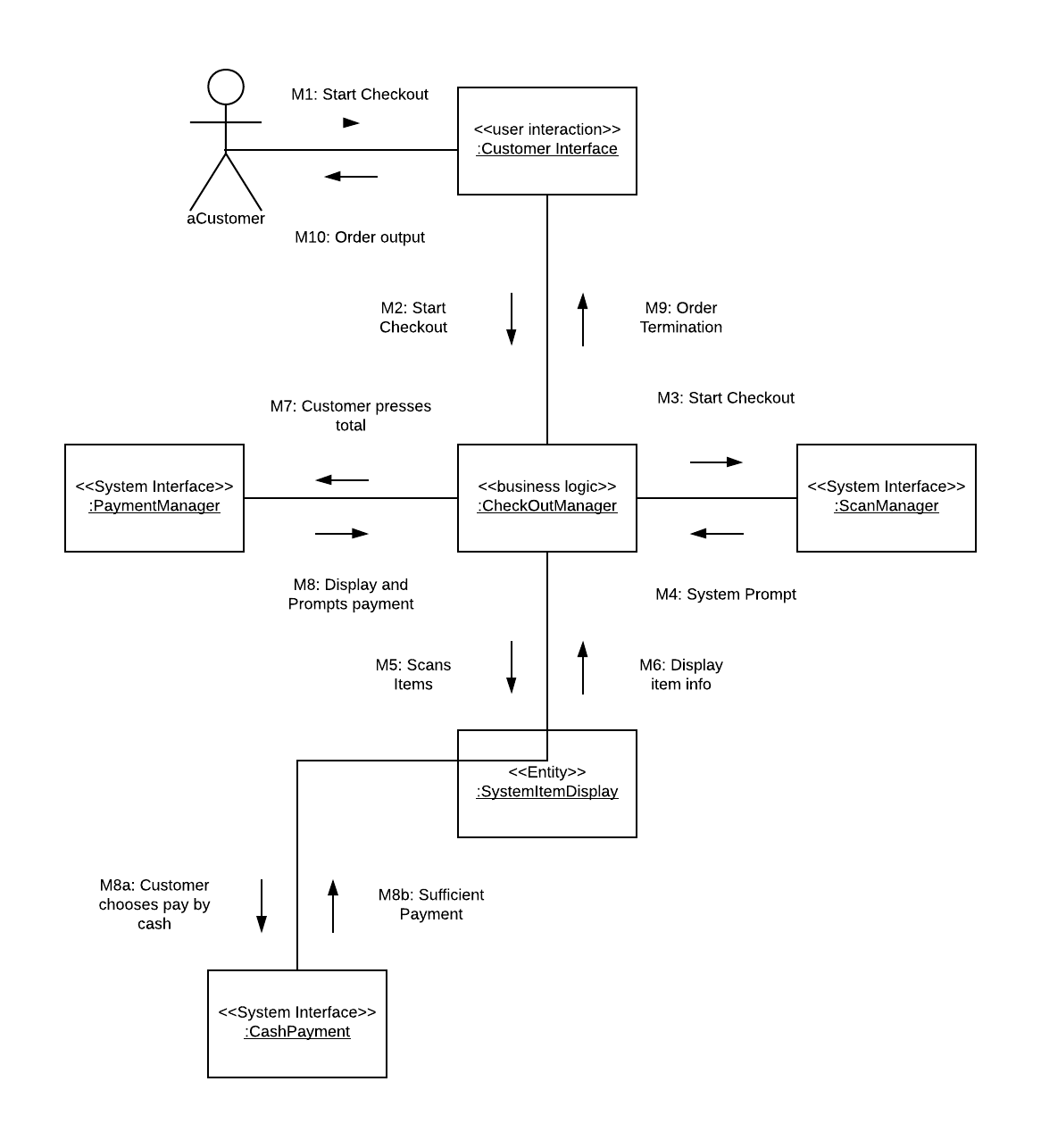
**Pay by Cash Main Sequence**



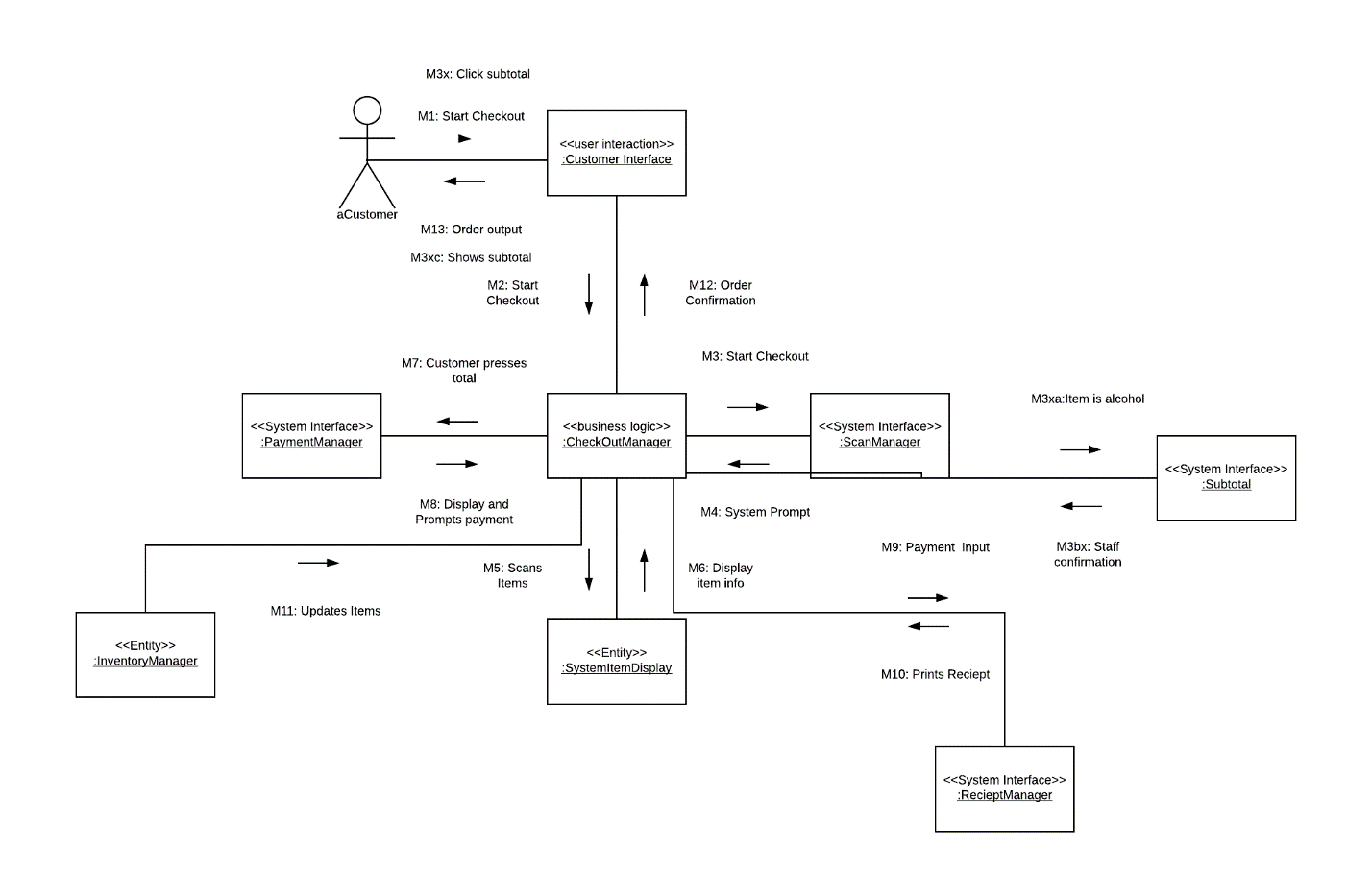
**Pay By Cash (Successful)**



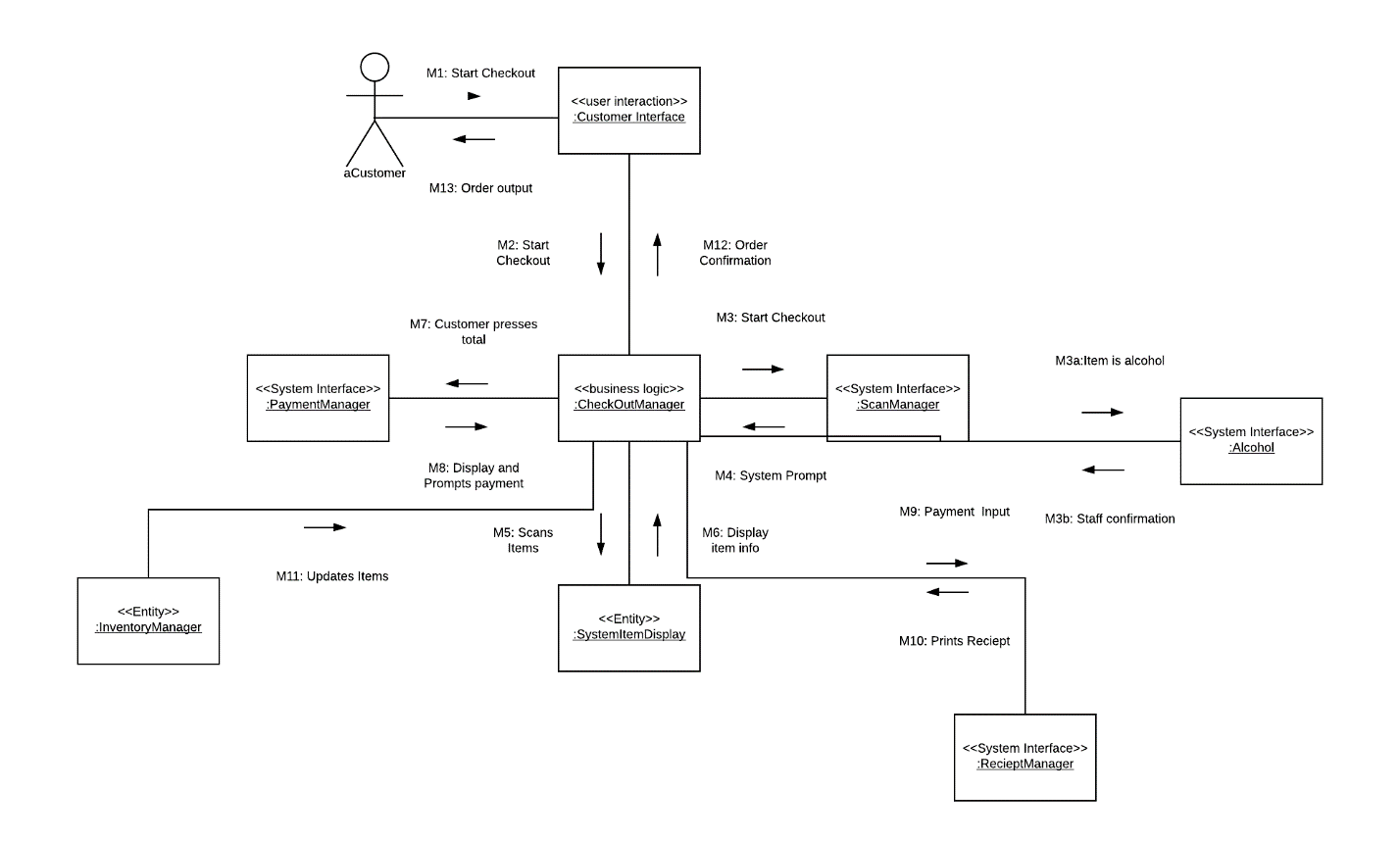
**Pay By Cash (Unsuccessful)**



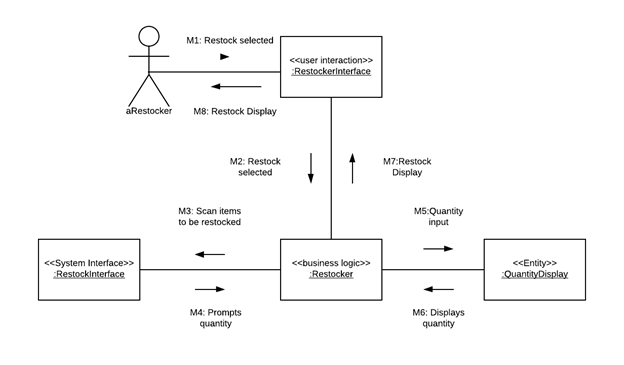
**Subtotal**



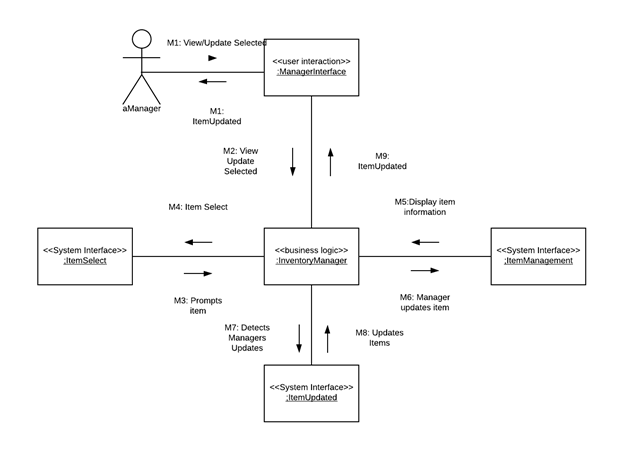
**Alcohol Present**



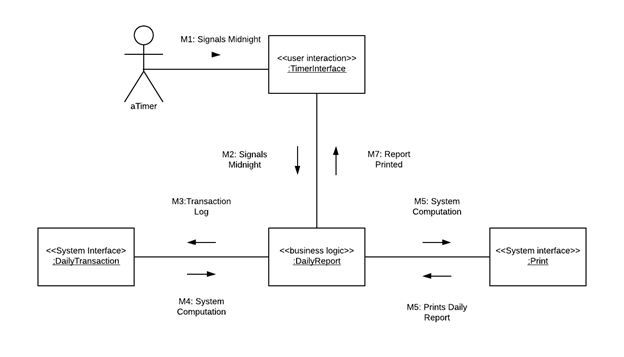
**Restock Product Inventory**



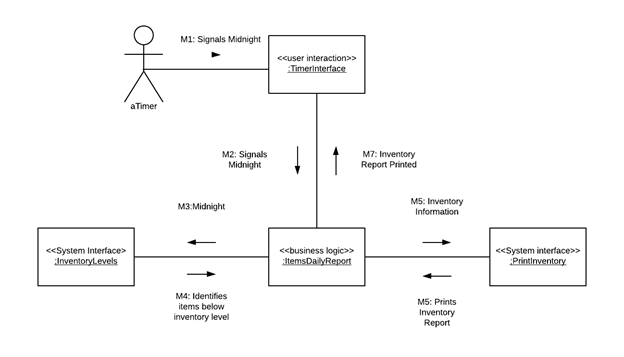
**View/Update Product Inventory**



**Print Daily Report**

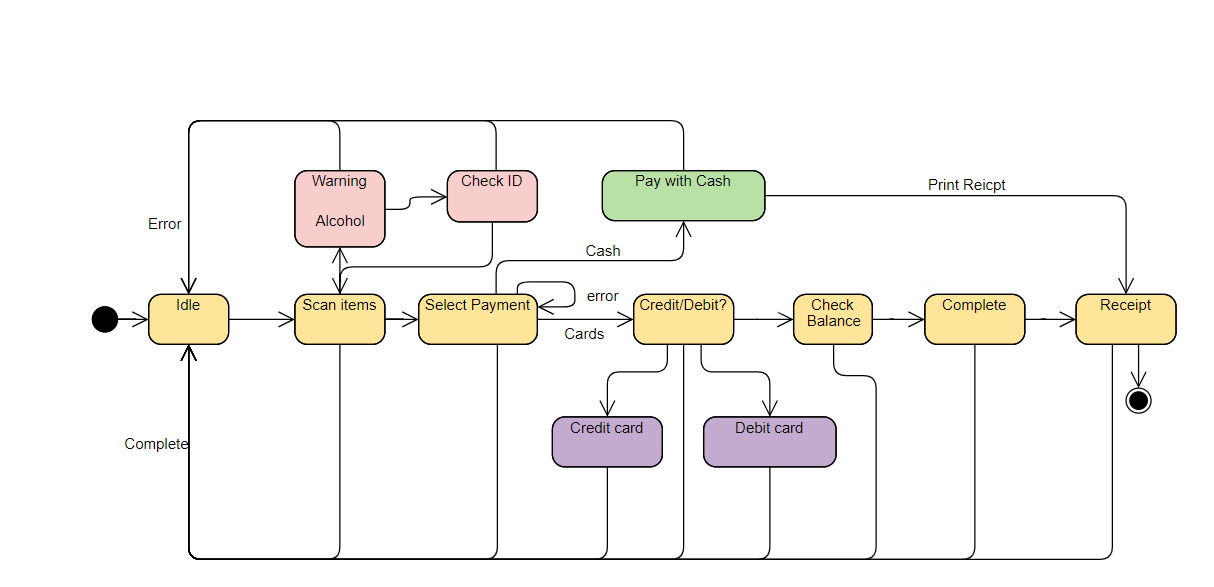


**Print Inventory Report**

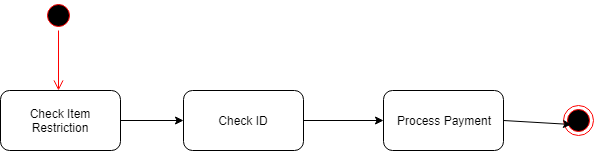


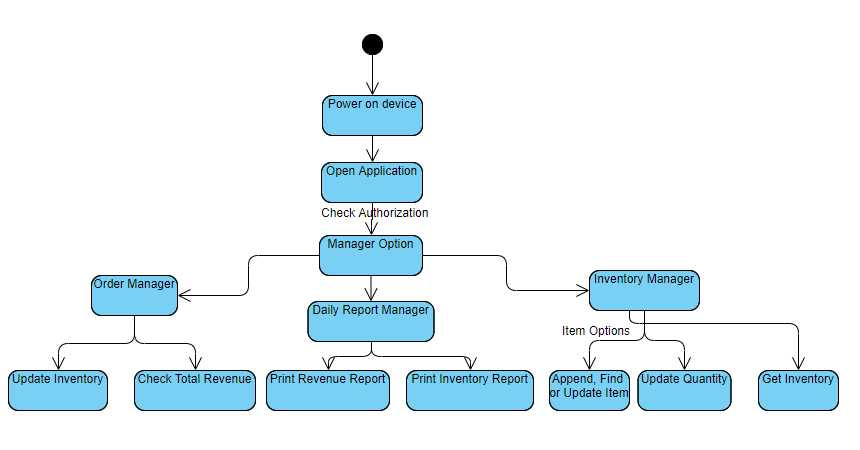
# 9. State Diagrams

**Making payment**



ID



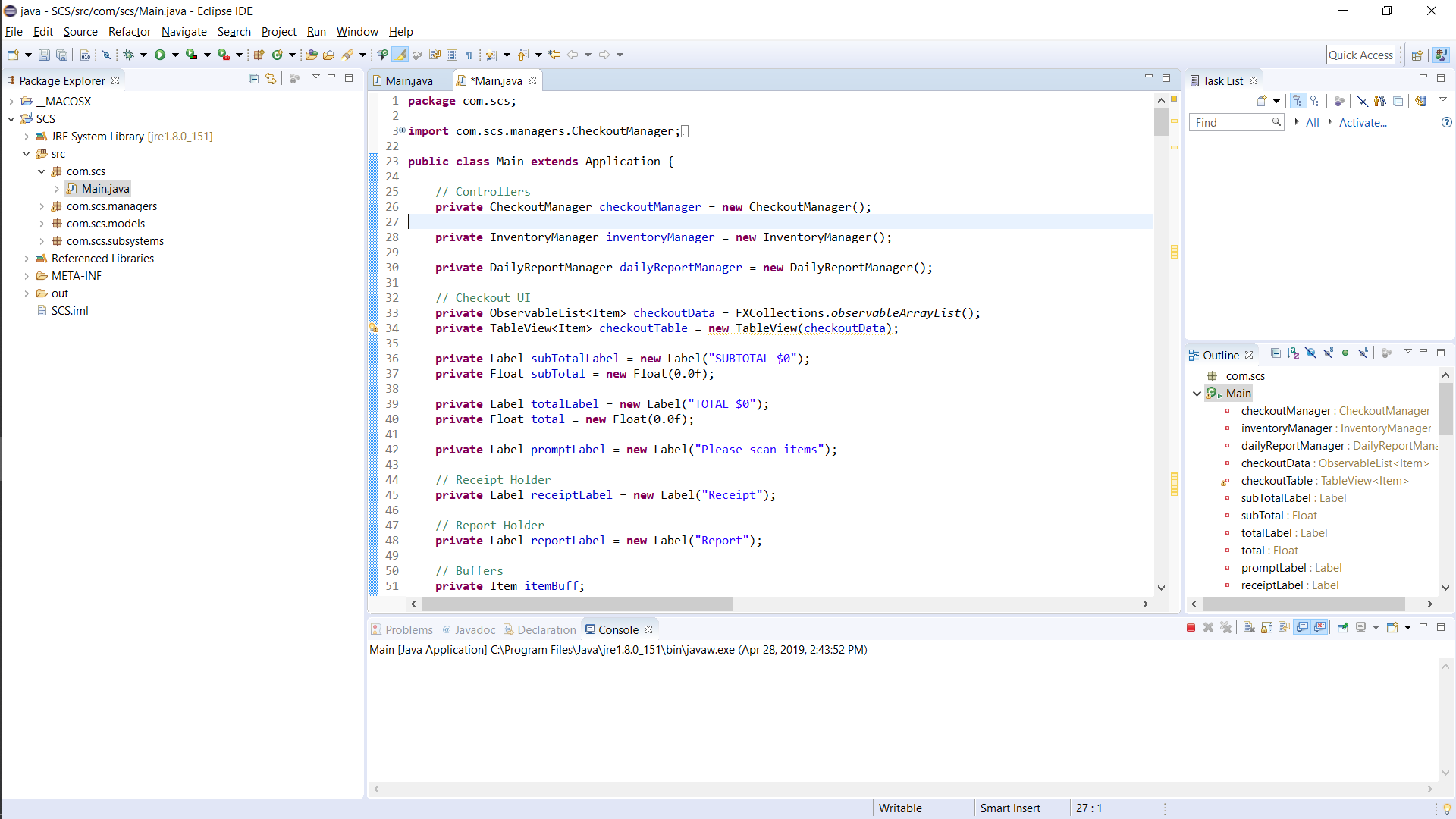
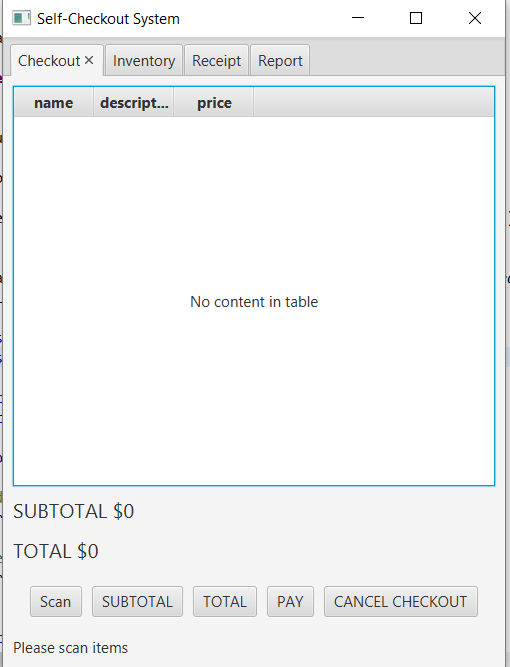
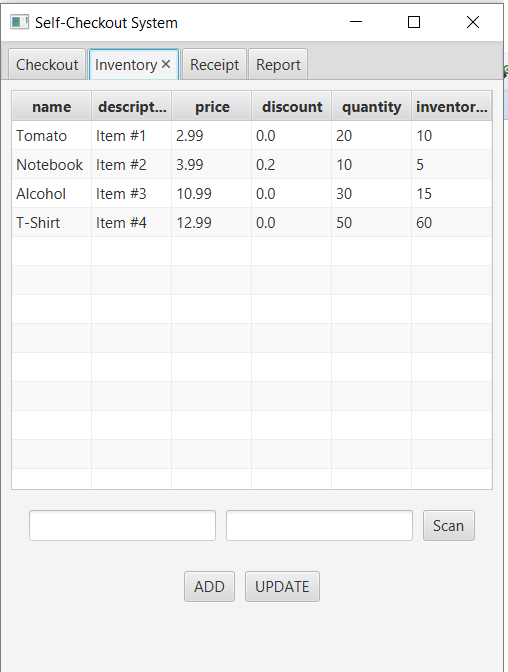
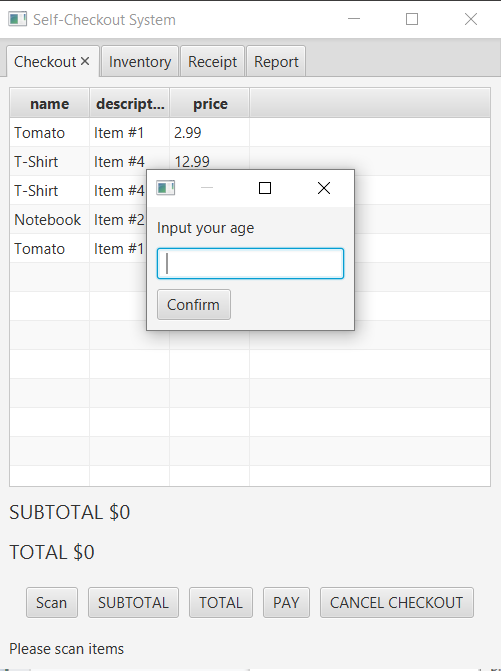
Manager Options:

# 10. Coding

See attachment file “java”.

11. User/Developer Guide

This project is compiled by Java, using Eclipse Java(IDE).

1. Opening Eclipse Java, choose to open file java.
2. Run X:\java\SCS\src\com\scs\main
3. 
4. **(Mobile app)**
5. There are 4 tabs for different users.
6. Customers can checkout through *Checkout* tab.
7. Staff and manager can update and view inventory through *Inventory* tab.**(Mobile app)**
8. Customers and Staff can view receipt through *Receipt* tab.
9. Managers can print daily report through *Report* tab.
10. By clicking Scan we can have a demonstration showing how it works. It will ask customers’ age for alcohol purchase.
11. **(Mobile app)**
12. Customers can check total and choose a payment for checkout.

# Appendix

* Mobile App(see part 11 guide)