Ridhima Joshi
Varshithanand Kotipalli
Mahrukh Mirza

SWEN5232
-
Software Construction
Assignment Number: 8

Professor: Dr. Findler
findler@uhcl.edu

Teaching Assistant:
Shail Panchal

HEB STORE

1. Brainstorm

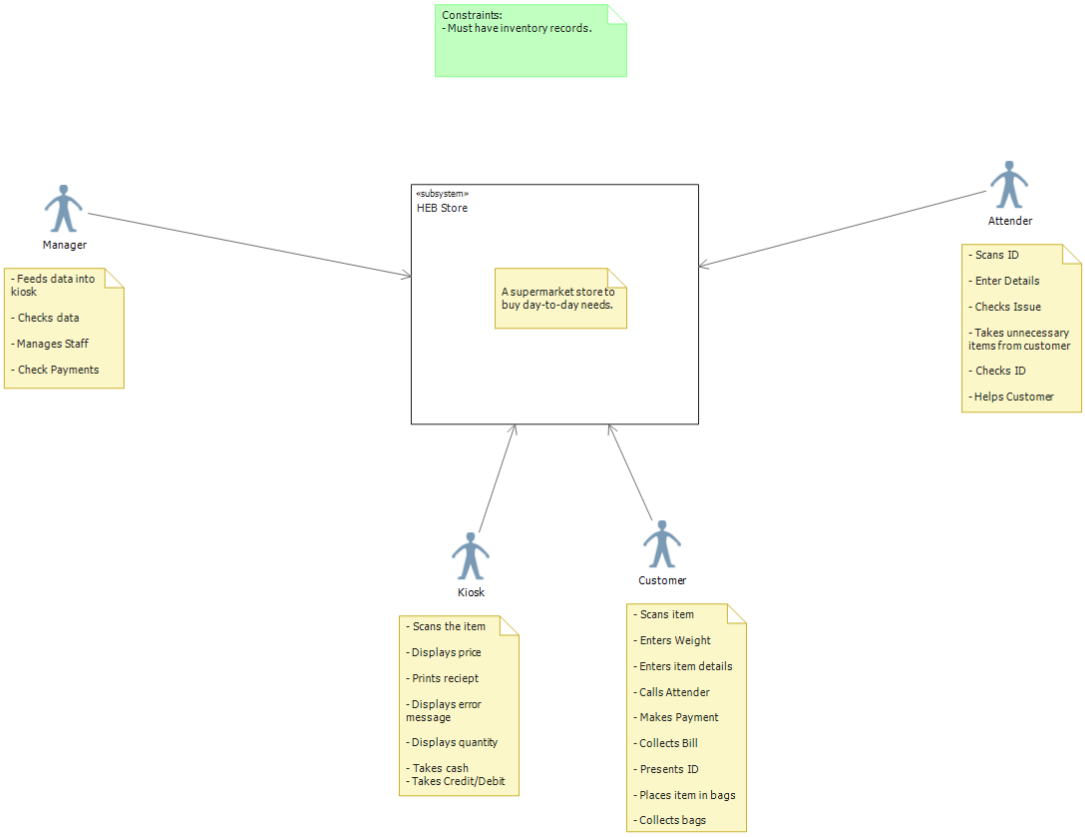

Figure 1: Brain Storm

2. Use Cases

2.1 Attender

This use case describes the activities the actor Attender performs while helping the customer at the automatic checkout machine.

Basic Flow:

1. The use case starts when the customer needs help with the machine.
2. The attender scans his/her ID card.
3. The attender enters his credentials.
4. The system provides access to him/her.
5. The attender checks the issue faced by the customer.
6. Performs Take unnecessary items from the customer task (S1).
7. Performs checking of customer ID (S2) to allow the customer to buy the alcohol.
8. The use case ends.

Sub flows:

S1. Take Unnecessary Items

1. The customer hands over the unwanted items.
2. Collect those items from the customer.
3. Use case flow resumed at the next step.

S2. Checking of Customer's ID

1. The customer buys alcohol.
2. Check the ID of the customer.
3. Use case flow resumed at the next step.

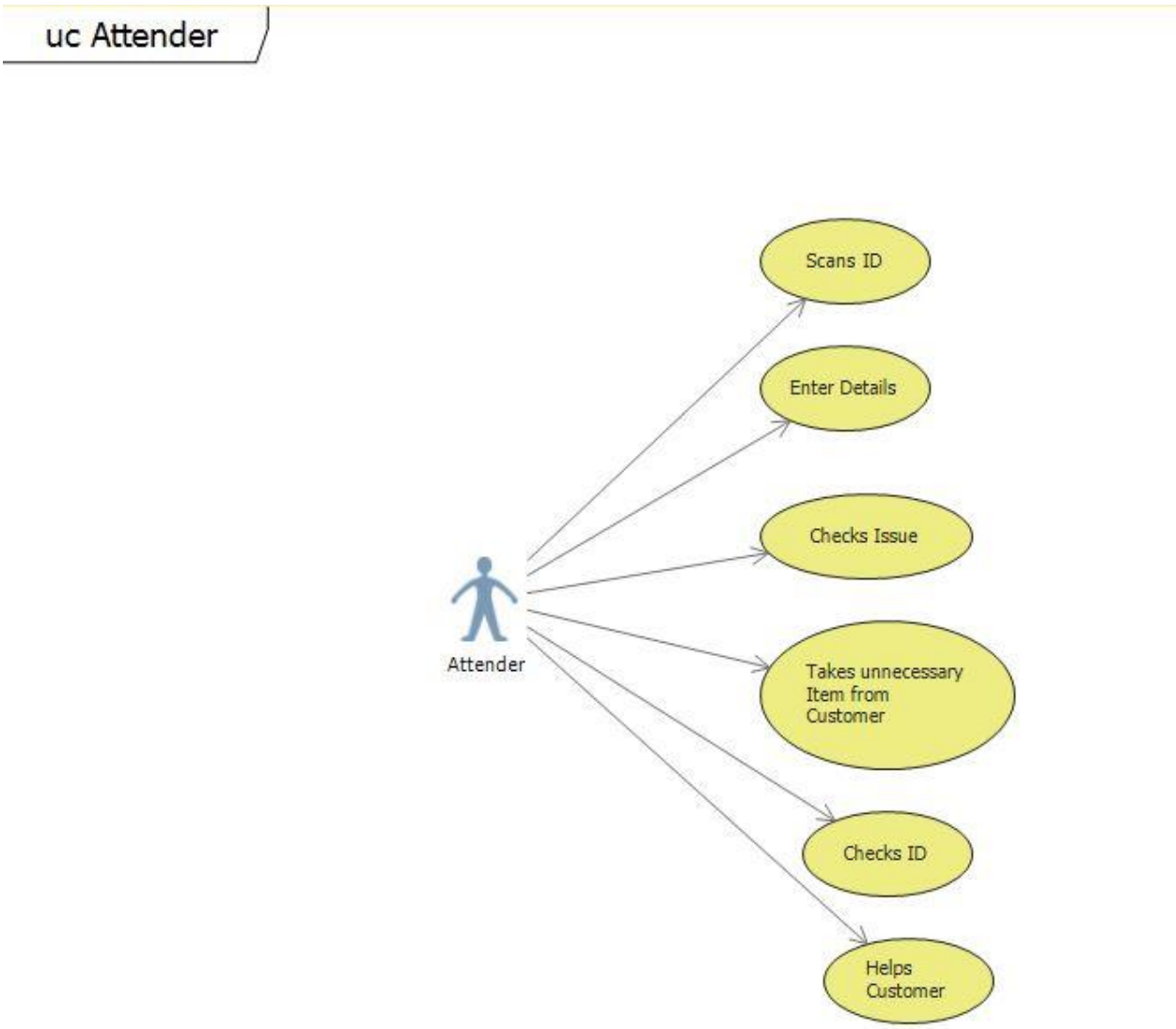

Figure 2: Use Case Diagram for Attender

2.2 Customer

This use case describes the activities the actor Customer performs while using the automatic checkout machine.

Basic Flow:

1. The use case starts when the customer is done with his/her shopping and needs to bill the items.
2. The customer scans the item using the item barcode.
3. The system scans the barcode.
4. The customer calls the attender (S1).
5. System displays the total bill amount.
6. Customer pays the amount.
7. System prints the copy of the bill.
8. Customer collects the copy of the bill.
9. Customer puts all the items in the bag.
10. Customer collects the bag.
11. The use case ends.

Sub flows:

S1. Calls the Attender

1. The customer scans the item.
2. System displays error in scanning.
3. Customer calls the attender.
4. Attender scans his/her ID.
5. Attender enters his/her details.
6. Attender enters the item manually.
7. Use case flow resumed at the next step.

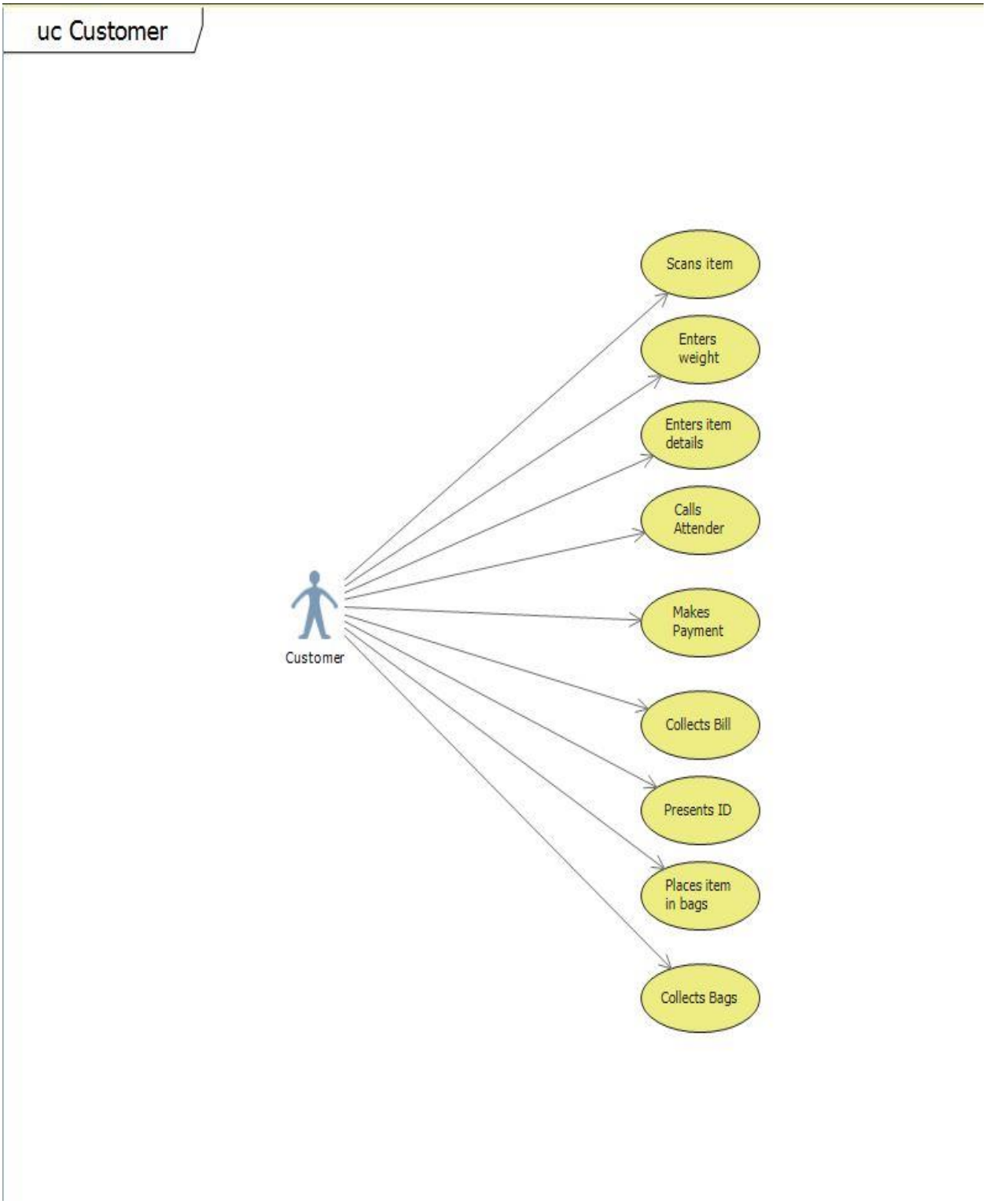

Figure 4: Use Case Diagram for Customer

Failure Scenario While Scanning an Item

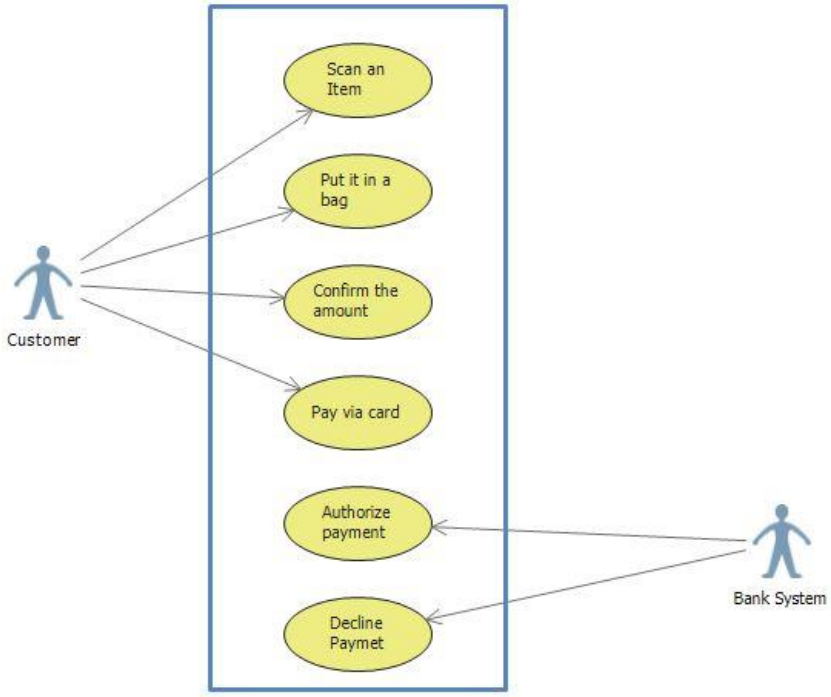

Figure 5: Use Case for failure Scenario While Scanning an Item

2.3 Kiosk

This use case describes the activities the actor Kiosk performs while the customer uses to bill the items.

Basic Flow:

1. The use case starts when the customer touches the screen to start the billing of an item.
2. The customer scans the item.
3. System reads the scanned item.
4. System displays error message (S1).
5. System displays the price.
6. System calculates the total.
7. Customer confirms the total.
8. Customer pays the amount.
9. System processes the payment.
10. System prints the bill.
11. The use case ends.

Sub flows:

S1. Displays Error Message

1. The customer scans the item.
2. System displays error in scanning.
3. Customer calls the attender.
4. Attender scans his/her ID.
5. Attender enters his/her details.
6. Attender enters the item manually.
7. Use case flow resumed at the next step.

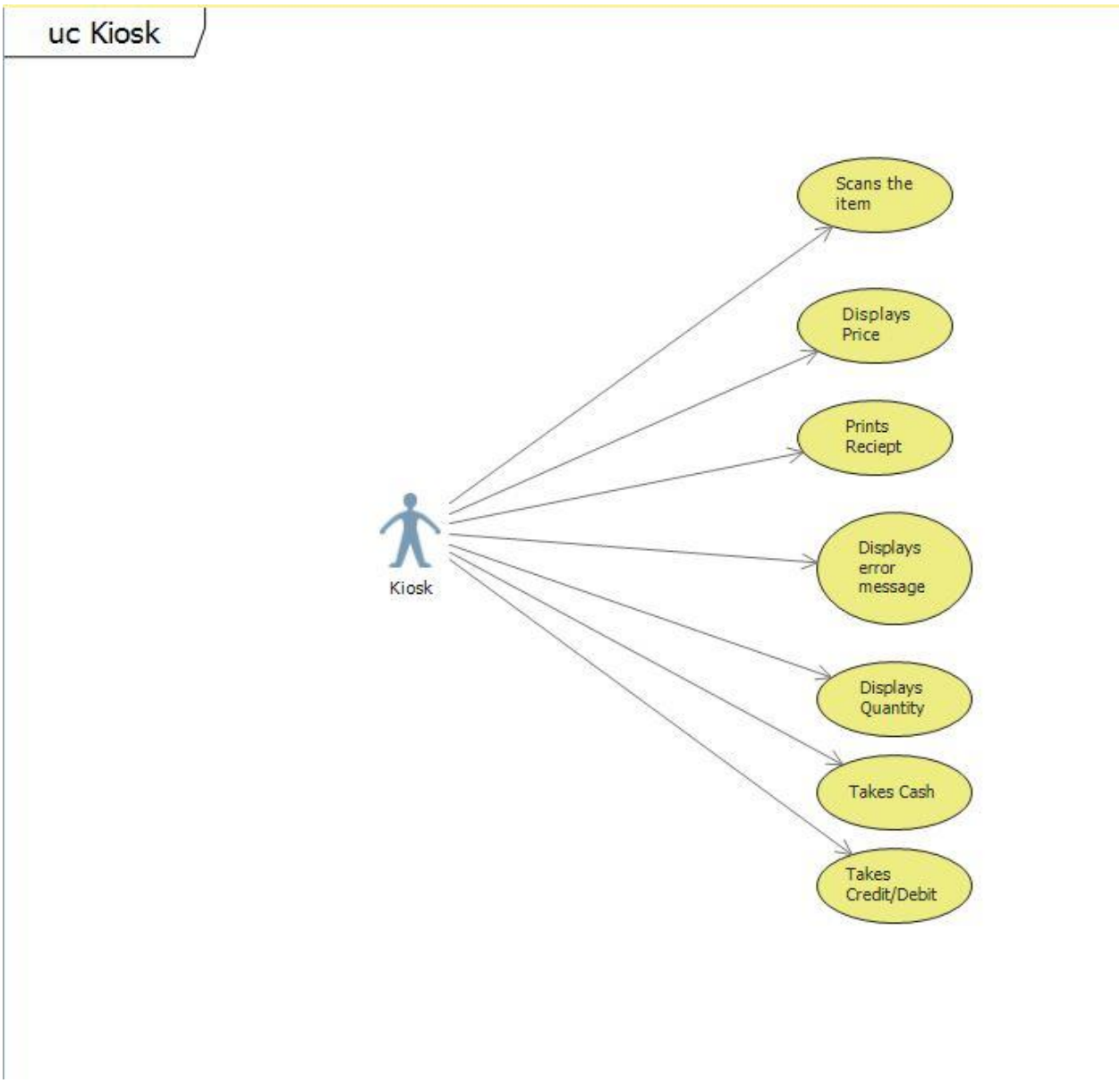

Figure 6: Use Case Diagram for Kiosk

2.4 Manager

This use case describes the activities the actor Manager performs.

Basic Flow:

1. The use case starts when the manager has feed in new data in the machine database.
2. The manager checks the data.
3. The manager manages staff (S1).
4. The checks payments.
5. The use case ends.

Sub flows:

S1. Manages Staff

1. The manager gives the employees lunch breaks.
2. Handles customer queries.
3. Use case flow resumed at the next step.

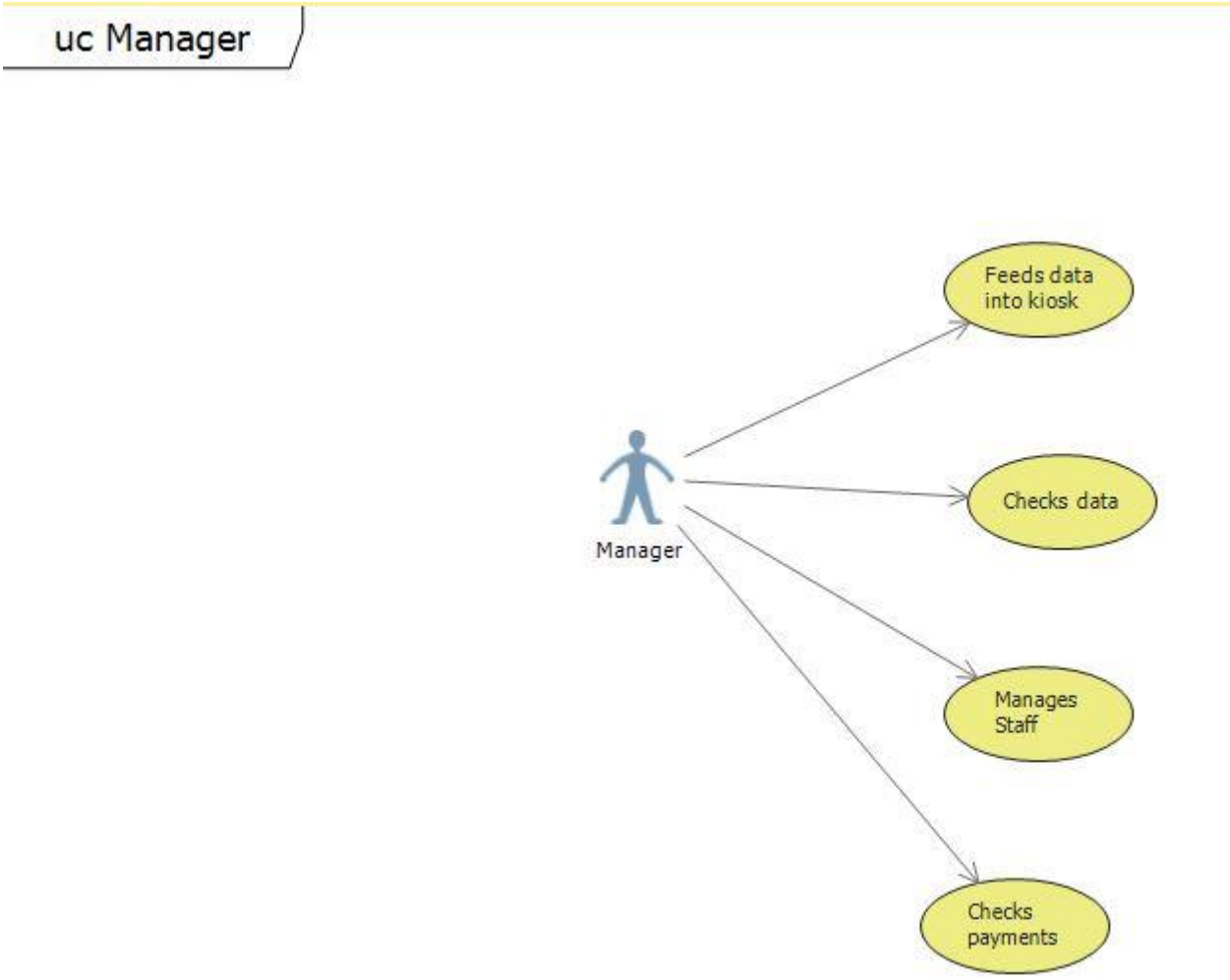

Figure 7: Use Case Diagram for Kiosk

3. Task Analysis

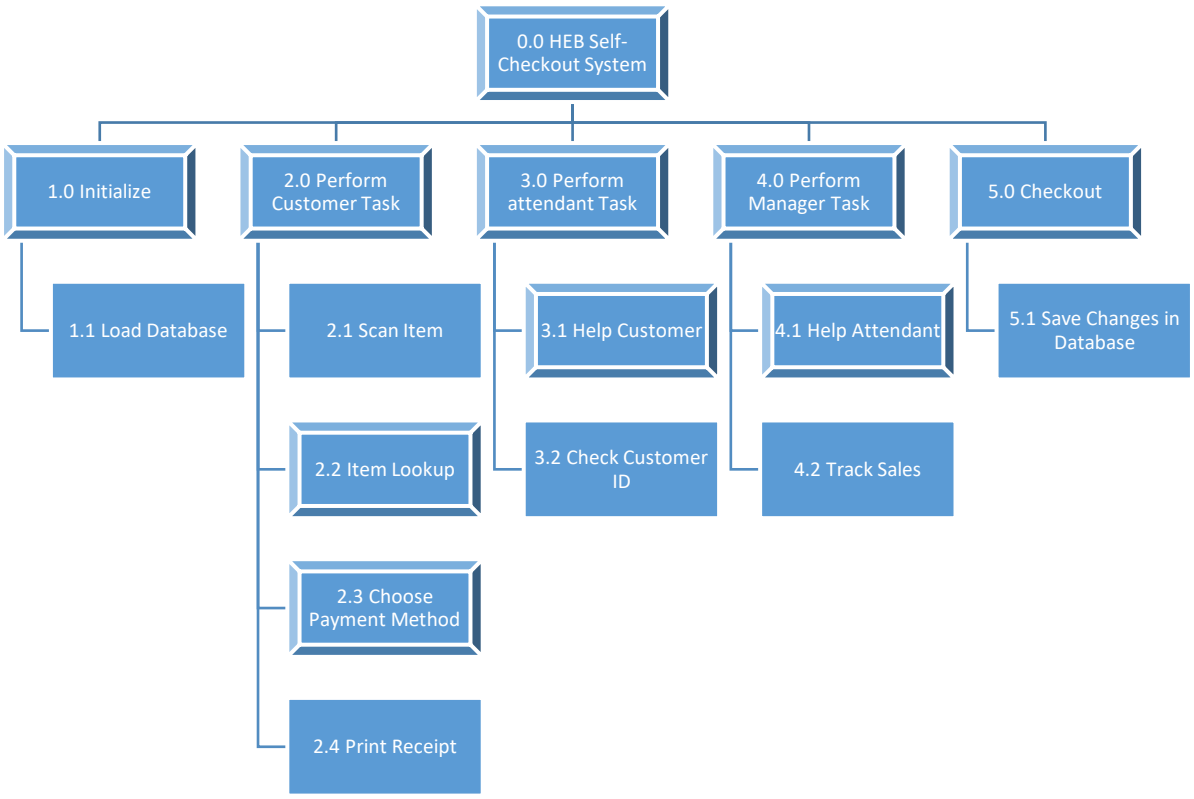

Figure 8: Task Analysis for Checkout System

0.0 HEB Self-Checkout System

1. What is the goal of this task?

The goal of this task is to help the customers check out easily at HEB.

2. What subtasks define this task?

The sub-tasks that define this task are

1.0 Initialize.

1.1 Load database

2.0 Perform customer task

2.1 Scan Items

2.2 Item Lookup

2.3 Choose Payment Method

2.4 Print Receipt

3.0 Perform attendant task

3.1 Help Customer

3.2 Check Customer ID

4.0 Perform Manager Tasks

4.1 Help Attendant

4.2 Track Sales

5.0 Checkout

5.1 Save Changes in Database

3. Is this task a subunit of a larger task?

No

4. What non-interface functions does this task require?

None

5. What kinds of inputs or actions does this task require from the user?

Items are supposed to be scanned by the customer

6. What kinds of outputs/results are expected by performing this task?

Easy checkout of items by using this self-checkout system

7. What automatic actions does this task expect from the system?

Display the prices of items that are scanned, calculate the total of items, accept the payment, print the receipt and save changes in the database

8. What special characteristics of this task should we record?

None

9. In this subtree, is there a task that must come before this one?

No

10. In this subtree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the tasks must be performed?

No

11. Which, if any, primary classes/entities are involved in this subtask?

Database, customers, items, scanner, monitor, billing machine

12. How can this task fail (or end in non-completion)?

This task can fail when the customer does not scan the items properly, when the system has timed out, when the payment is not accepted and when the item details are not available in the system

13. How frequently is this task performed?

Whenever the customer uses this self-checkout system

14. How open is this task, especially in terms of its sequence or inputs?

This task is not open

15. What if any are the specific usability expectations (e.g. ease of use) for this task and how do we anticipate determining if we have satisfied the user expectations? (Hint: like ease of use, ease of learning. As you specify this, be specific as to what "ease of use" and other usability expectations mean and how you will evaluate these.)

Ease of use

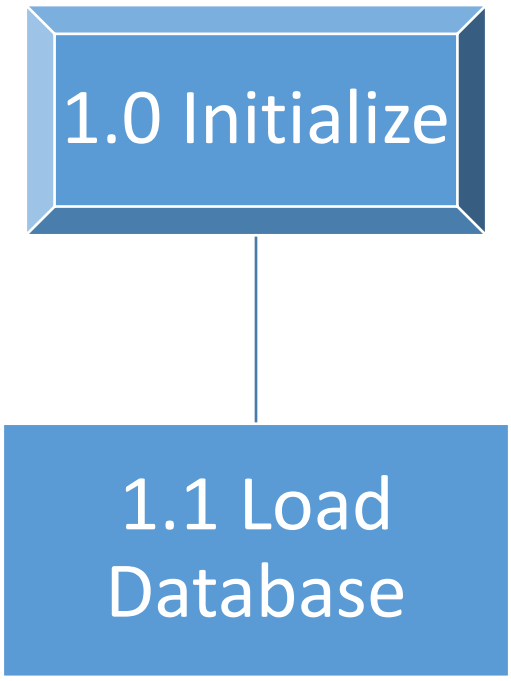

Figure 9: Task Analysis for Initialize

1,0 Initialize

1. What is the goal of this task?

The goal of this task is to initialize the database of the self-checkout system at HEB so that the customer can easily use the system

2. What subtasks define this task?

The sub-tasks that define this task are

1.1 Load database

3. Is this task a subunit of a larger task?

Yes, this task is a subunit of larger task i.e HEB Self-checkout System

4. What non-interface functions does this task require?

None

5. What kinds of inputs or actions does this task require from the user?

The customer should tap the screen

6. What kinds of outputs/results are expected by performing this task?

The customer can view the initial screen and can start the scanning of items

7. What automatic actions does this task expect from the system?

The system should load the database at some location

8. What special characteristics of this task should we record?

None

9. In this subtree, is there a task that must come before this one?

Yes, task 0.0 HEB Self-checkout system

10. In this subtree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the tasks must be performed?

Yes, there is a specific sequence in which this task is performed.

11. Which, if any, primary classes/entities are involved in this subtask?

Database, customers, items, scanner, monitor, billing machine

12. How can this task fail (or end in non-completion)?

This task can fail when the system is not initialized properly

13. How frequently is this task performed?

Whenever the customer uses this self-checkout system

14. How open is this task, especially in terms of its sequence or inputs?

This task is not open

15. What if any are the specific usability expectations (e.g. ease of use) for this task and how do we anticipate determining if we have satisfied the user expectations? (Hint: like ease of use, ease of learning. As you specify this, be specific as to what "ease of use" and other usability expectations mean and how you will evaluate these.)

None.

1.1 Load Database

1. What is the goal of this task?

The goal of this task is to load the database so that the customer is able to checkout with ease using the HEB Self-checkout system.

2. What subtasks define this task?

None

3. Is this task a subunit of a larger task?

Yes. This task is a subunit of initialize.

4. What non-interface functions does this task require?

None

5. What kinds of inputs or actions does this task require from the user?

The customer should tap the screen for the initial options of the screen to be displayed

6. What kinds of outputs/results are expected by performing this task?

Customer sees the initial screen and can now scan the items

7. What automatic actions does this task expect from the system?

None

8. What special characteristics of this task should we record?

None

9. In this subtree, is there a task that must come before this one?

No

10. In this subtree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the tasks must be performed?

No

11. Which, if any, primary classes/entities are involved in this subtask?

Database, customers, items, scanner, monitor, billing machine

12. How can this task fail (or end in non-completion)?

This task can fail if the server is down and the database could not load properly

13. How frequently is this task performed?

Whenever the customer uses this self-checkout system

14. How open is this task, especially in terms of its sequence or inputs?

This task is not open

15. What if any are the specific usability expectations (e.g. ease of use) for this task and how do we anticipate determining if we have satisfied the user expectations? (Hint: like ease of use, ease of learning. As you specify this, be specific as to what "ease of use" and other usability expectations mean and how you will evaluate these.)

None

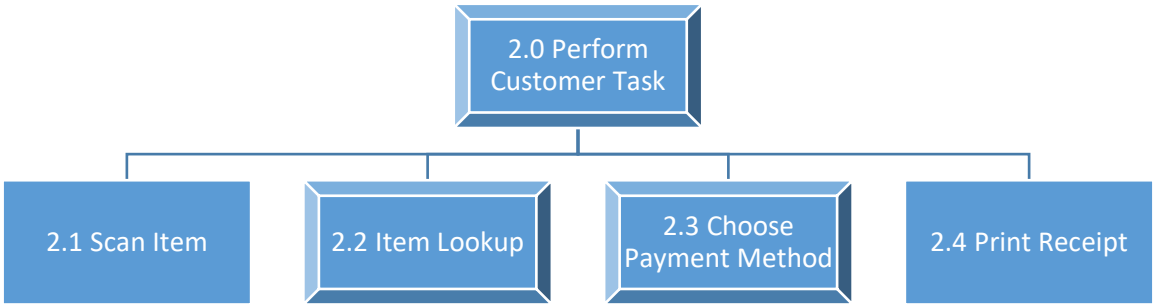

Figure 10: Task Analysis for Customer Task

2.0 Perform Customer Task

1. What is the goal of this task?

The goal of this task is to perform all the tasks of the customer. The tasks include scanning the item, item lookup, choosing the payment method and printing the receipt

2. What subtasks define this task?

2.1 Scan Item

2.2 Item Lookup

2.3 Choose Payment Method

2.4 Print Receipt

3. Is this task a subunit of a larger task?

Yes. This task is a subunit of HEB Self-Checkout System.

4. What non-interface functions does this task require?

None

5. What kinds of inputs or actions does this task require from the user?

The customer should tap the screen for the initial options of the screen to be displayed

6. What kinds of outputs/results are expected by performing this task?

Customer should be able to perform the scanning of item, easily looking up for items, should easily be able to choose payment method and get the receipt

7. What automatic actions does this task expect from the system?

None

8. What special characteristics of this task should we record?

None

9. In this subtree, is there a task that must come before this one?

No

10. In this subtree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the tasks must be performed?

Yes. Scan the items, item lookup, choose payment method and print receipt

11. Which, if any, primary classes/entities are involved in this subtask?

Database, customers, items, scanner, monitor, billing machine

12. How can this task fail (or end in non-completion)?

This task can fail if the system has timed out, if the item is not scanned properly or if the payment is not accepted

13. How frequently is this task performed?

Whenever the customer uses this self-checkout system

14. How open is this task, especially in terms of its sequence or inputs?

This task is not open

15. What if any are the specific usability expectations (e.g. ease of use) for this task and how do we anticipate determining if we have satisfied the user expectations? (Hint: like ease of use, ease of learning. As you specify this, be specific as to what "ease of use" and other usability expectations mean and how you will evaluate these.)

None

2.1 Scan Items

1. What is the goal of this task?

The goal of this task is that the customer should be able to scan the items properly without any failure

2. What subtasks define this task?

None

3. Is this task a subunit of a larger task?

Yes. This task is a subunit of perform customer task.

4. What non-interface functions does this task require?

None

5. What kinds of inputs or actions does this task require from the user?

The customer should get the items ready to scan and must scan them

6. What kinds of outputs/results are expected by performing this task?

Customer should be able to scan the items and the total amount of money should be displayed

7. What automatic actions does this task expect from the system?

None

8. What special characteristics of this task should we record?

None

9. In this subtree, is there a task that must come before this one?

No

10. In this subtree, is there a task for which this one is required to be immediately preceding? Is there any specific sequence in which the tasks must be performed?

Yes. Choose payment method and print receipt

11. Which, if any, primary classes/entities are involved in this subtask?

Database, customers, items, scanner, monitor, billing machine

12. How can this task fail (or end in non-completion)?

This task can fail if the item is not properly scanned

13. How frequently is this task performed?

Whenever the customer uses this self-checkout system

14. How open is this task, especially in terms of its sequence or inputs?

This task is not open

15. What if any are the specific usability expectations (e.g. ease of use) for this task and how do we anticipate determining if we have satisfied the user expectations? (Hint: like ease of use, ease of learning. As you specify this, be specific as to what "ease of use" and other usability expectations mean and how you will evaluate these.)

None

4. Business Model

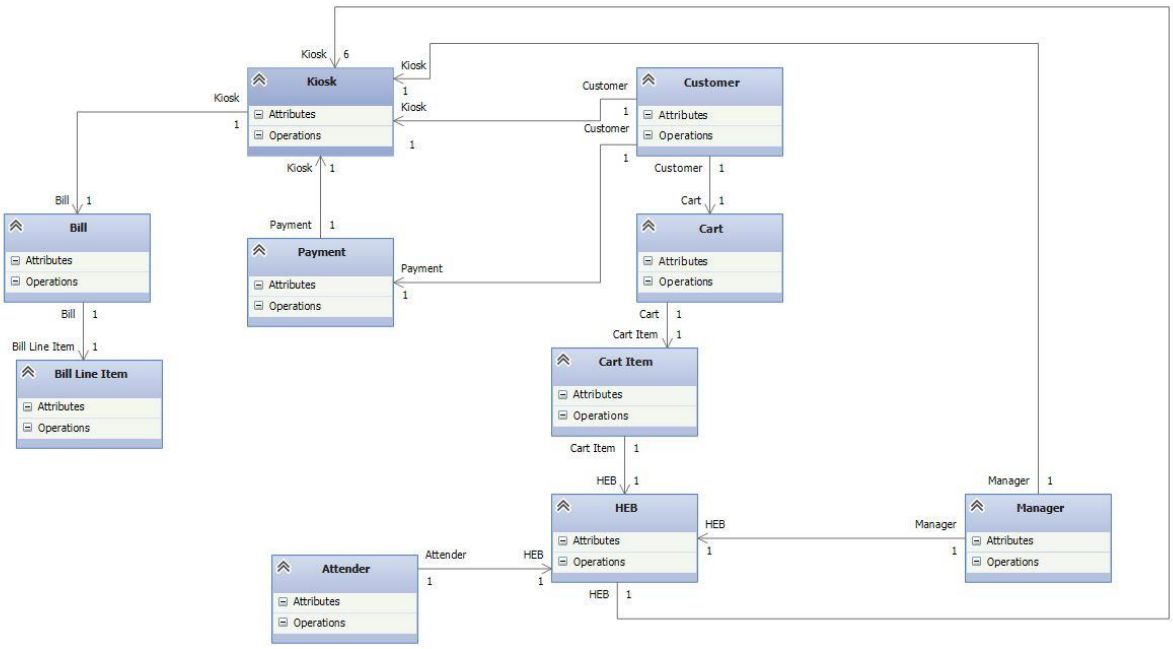

Figure 12: Domain Model

5. Activity Diagram

1. Attender

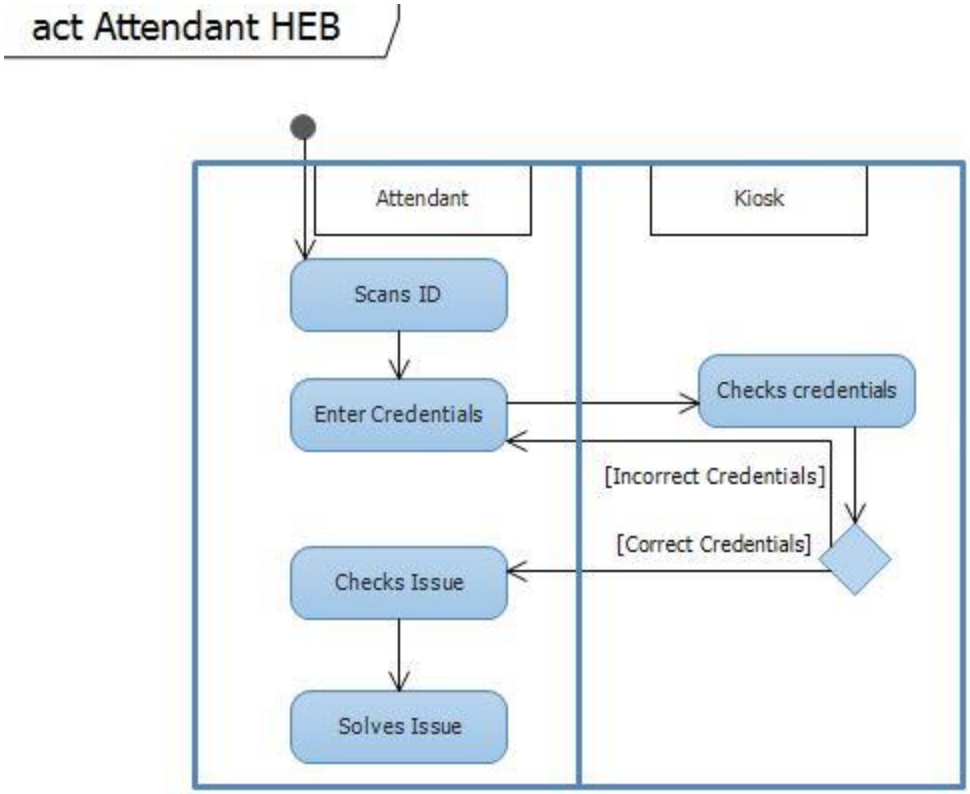

Figure 13: Activity Diagram of Attender

2. Customer

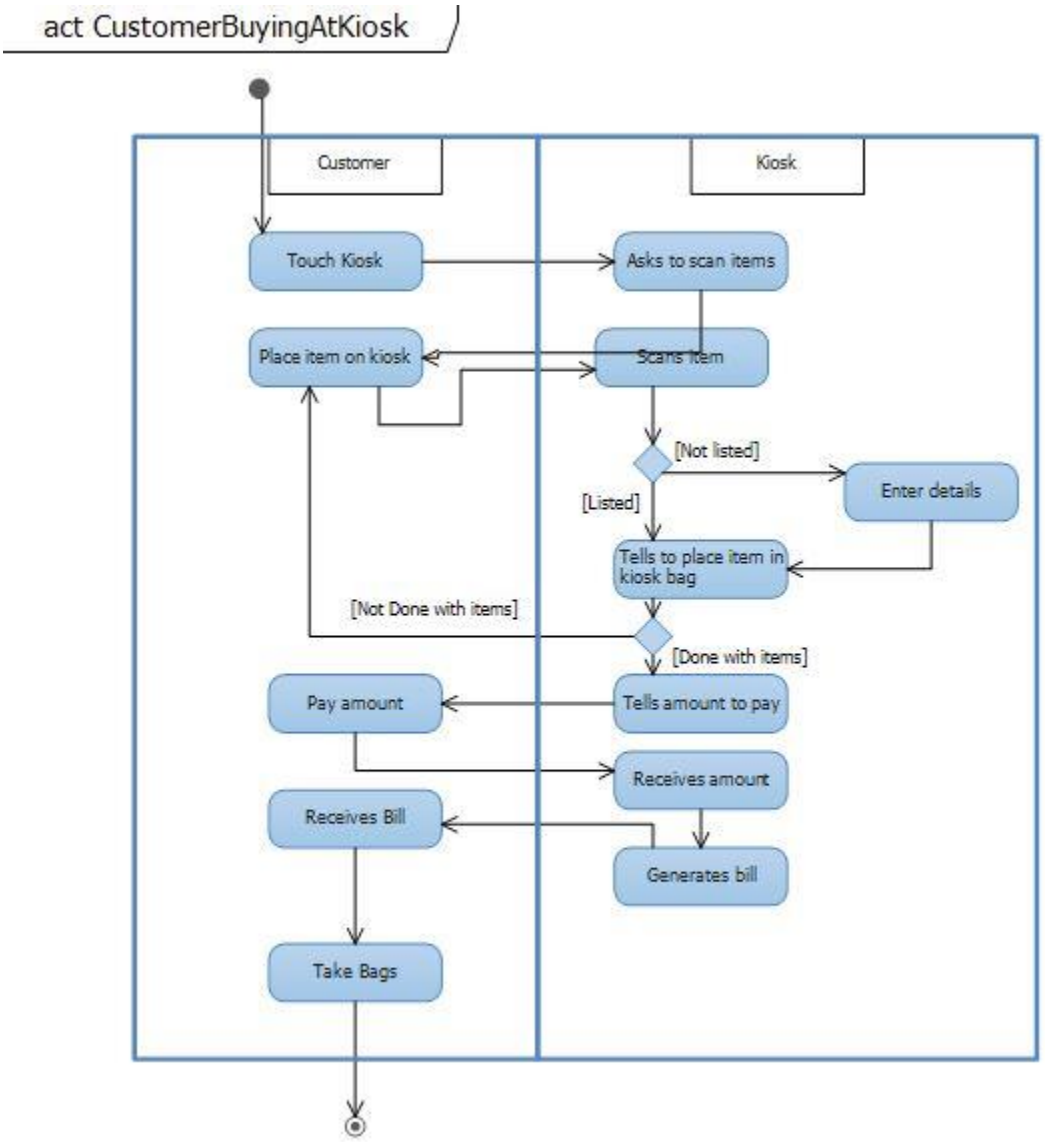

Figure 14: Activity Diagram of Customer

3. Purchasing Alcohol

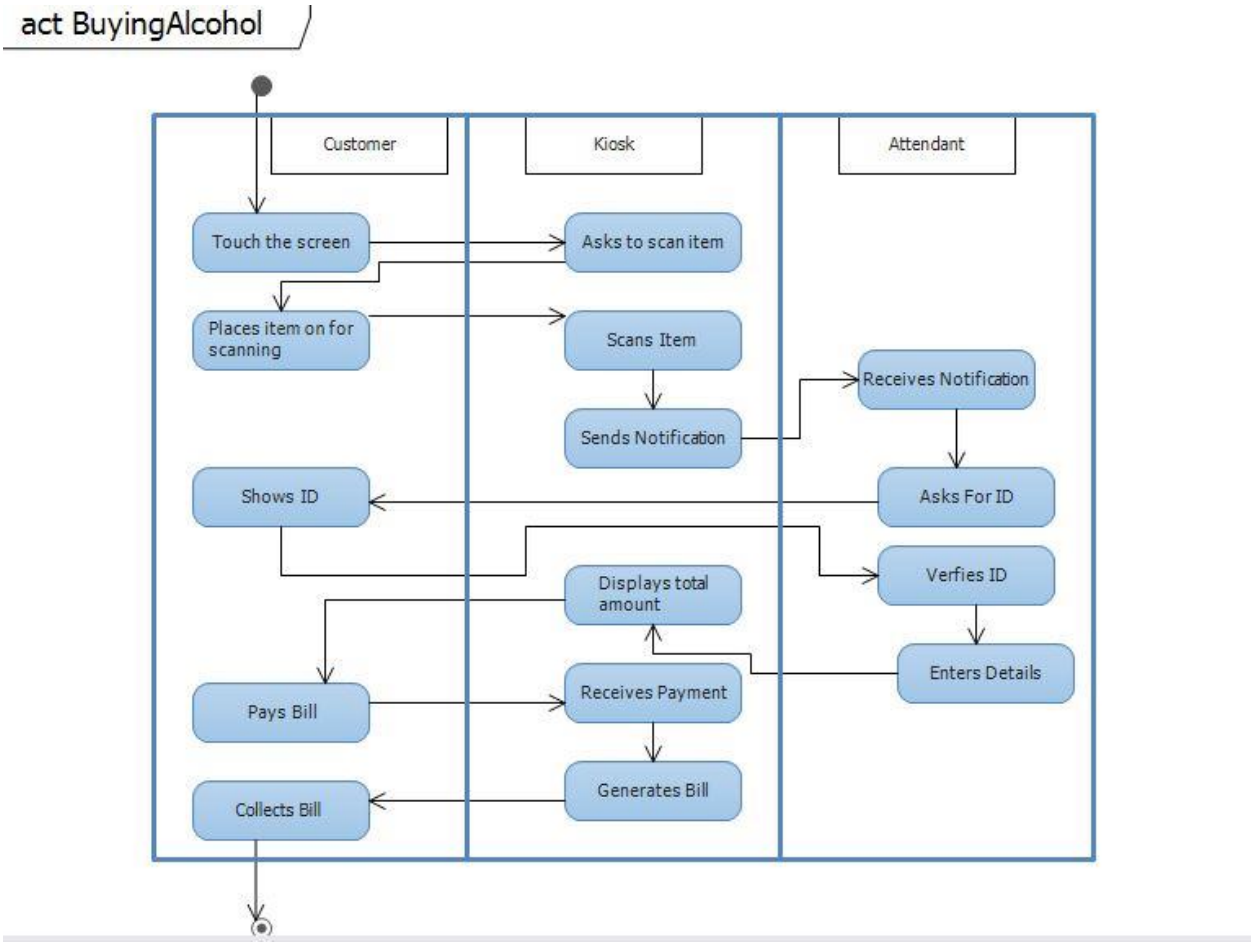

Figure 15: Activity Diagram of purchasing Alcohol

6. State Diagram

Customer

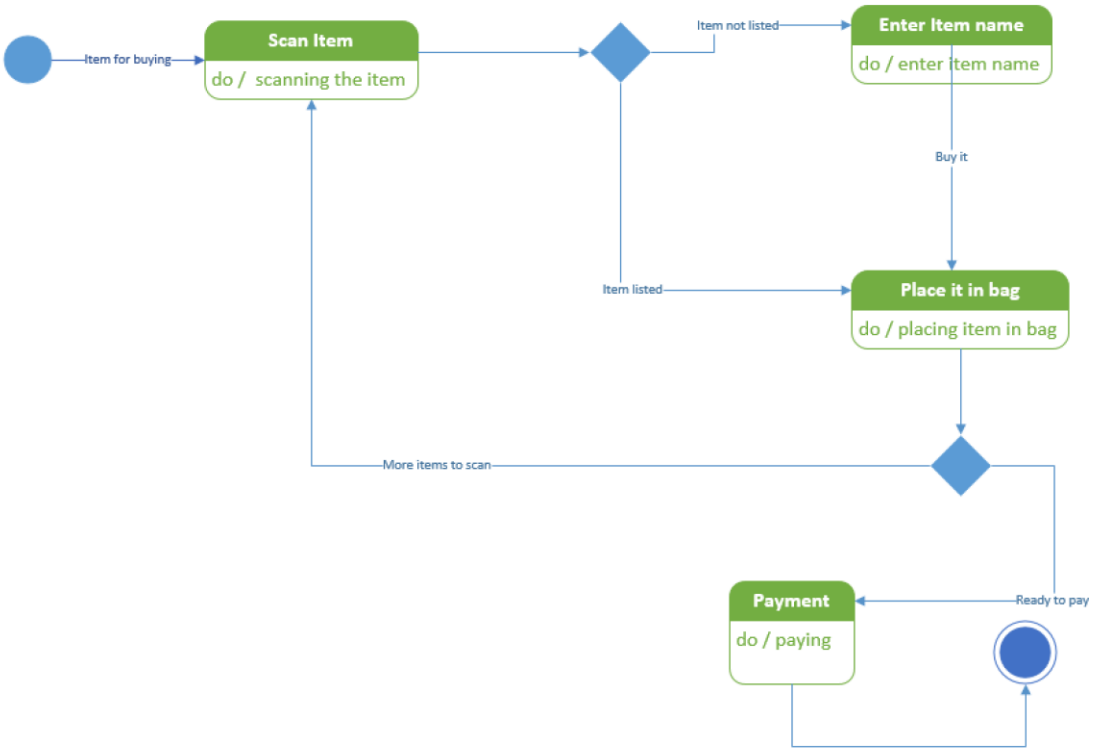

Figure 16: State Diagram of customer at the Kiosk

Kiosk

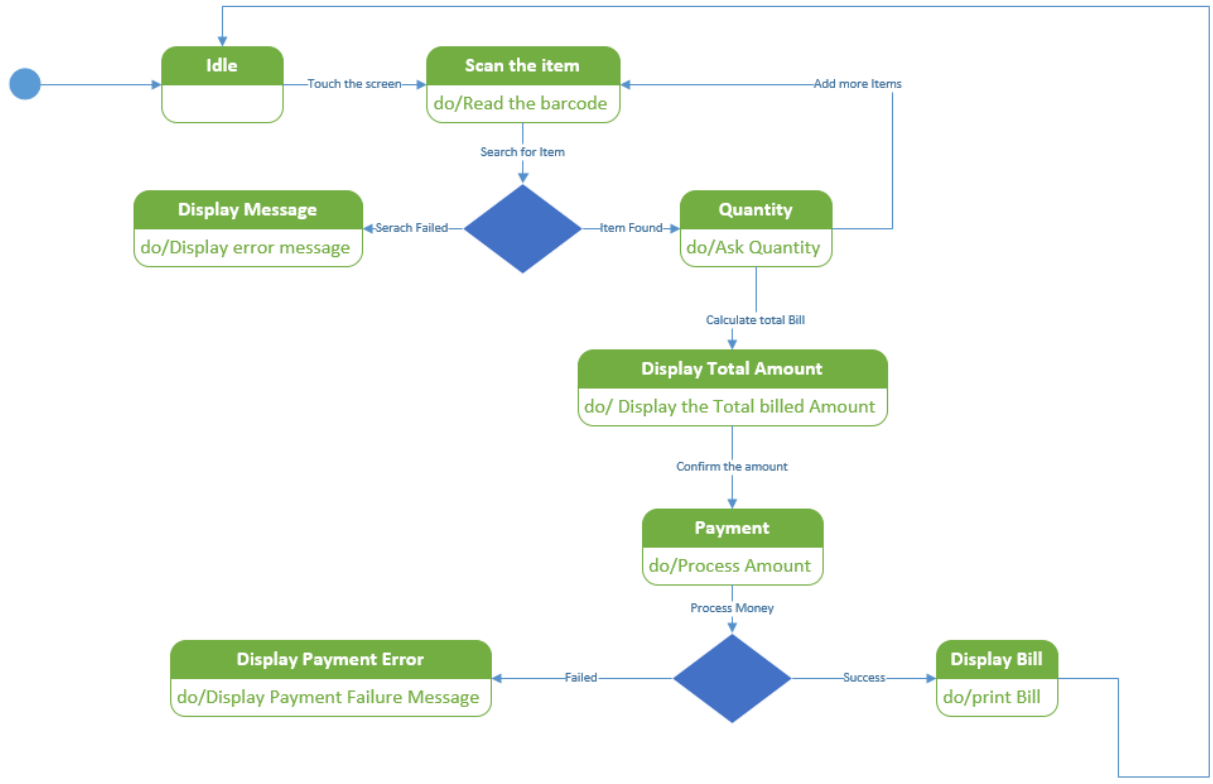

Figure 17: State Diagram of HEB Kiosk

7. Architecture Diagram

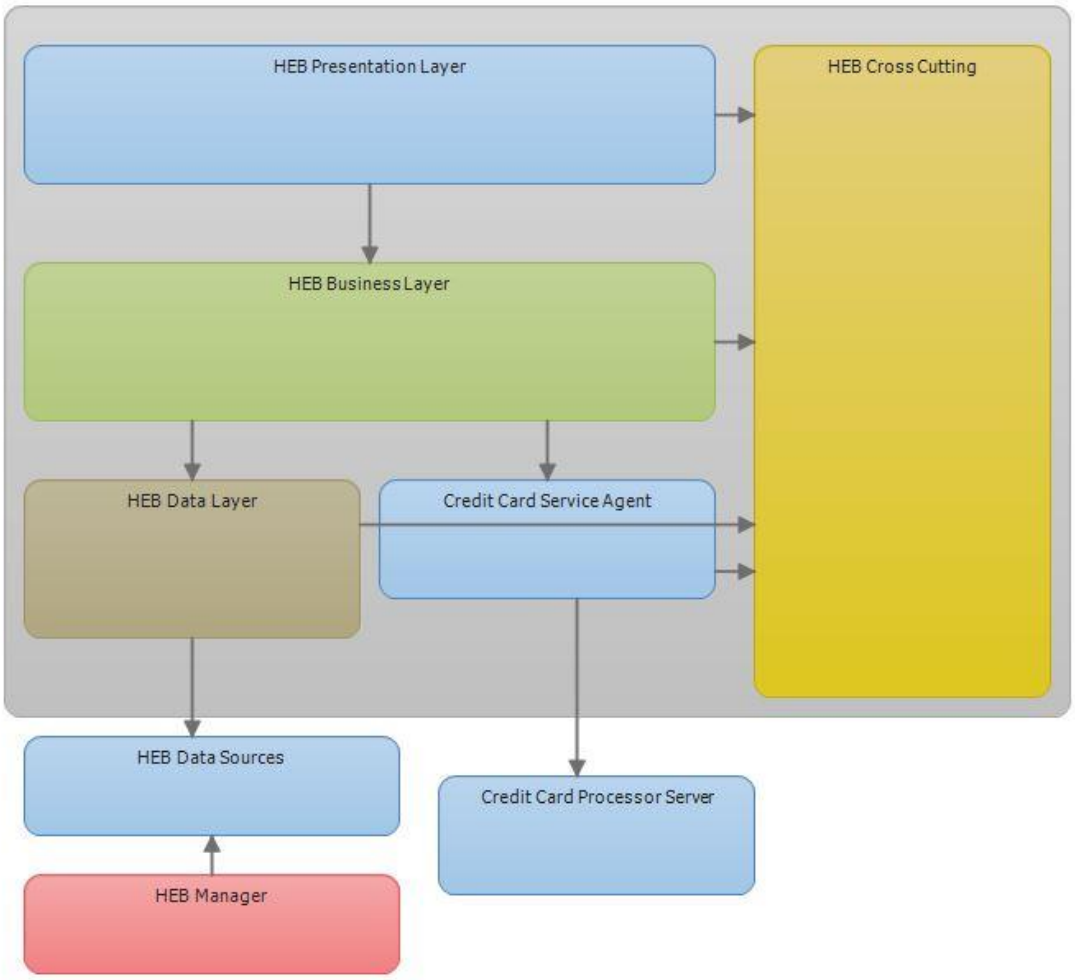

Figure 18: Architecture Diagram