Part I The Use Case Model

How many times have you reached the check-out in a store to see lines of shoppers waiting to pay for their goods, and considered just leaving your items in the store and coming back when it is quieter, or even going somewhere else?

How many times a week do you arrive at the check-out with a full cart, which you must unload onto the conveyor or self-checkout (scanning every item as you unload it from the cart). Or a store assistant re-loads your cart differently than how you wanted it, so that you have to load and organize your cart a second time?

Have you ever reached the checkout of a store and received a bill that is for a different amount than what you were expecting?

How many times have you find yourself searching for an item in a retail store with no store assistant visible to assist you?

These scenarios recur millions of times every day throughout the US alone. If you have you ever thought to yourself: 'There must be a more efficient way of doing this' then, consider the following business vision:

Consider construction of a computer system that vastly improves the customer experience in the store, by reducing the time spent shopping and assisting the customer to fulfill their shopping requirements.

- A system that allows customers to keep a running total cost of all items added to their shopping basket.
- A system that allows customers to pay for the items in their shopping basket from anywhere in the store.
- A system that allows the store to confirm that items leaving the store have been paid for.
- A system that allows a customer to receive assistance from any location in the store, without having to search for a store assistant.
- A system that tracks customer purchases and makes recommendation based upon their shopping habits.
- A system that allows customers to create a list of shopping items and assists the customer with fulfilling that list as they shop.

1 Overview

This article contains the results of a business analysis of the shopping cart problem described in the above vision statement.

The analysis uses the UML notation and RUP-like process that is described in my book Analysis Through Pictures.

The chosen business problem is one that all readers are able to associate with through personal experience.

1.1 Purpose

The purpose of this article is to demonstrate the role of the Business Analyst when considering improvement and automation of a business process.

1.2 Goals

The goals are stated in terms of Problem Statements which are derived from a project vision.

The problem of	Takes too long to check out.
Affects	Shoppers waiting to pay for items in their shopping cart.
The impact of which is	Long lines, crowding in the store, frustrated shoppers, loss of revenue through abandoned shopping carts.
A successful solution would be	Reduce checkout lines by having items in the shopping cart identified before reaching the checkout.

The problem of	Loading and unloading the cart too many times.
Affects	Shoppers using carts loaded with items from the store.
The impact of which is	Increases time to checkout, additional work for shoppers; increased chance of damaged goods.
A successful solution would be	Require 1 load and 1 unload of the shopping cart by removing the need to unload in order to scan the items in the shopping cart.

The problem of	Do not know how much a cart of items is going to cost until checkout occurs.
Affects	Shoppers loading carts with items.
The impact of	Shopper may overspend their budget without realizing it until checkout,

which is	causing delays at the checkout counter and additional work for store workers.
A successful solution would be	Keep the shopper continuously informed of the cost of their cart while they are shopping, so they know how much to expect to pay at the checkout.

The problem of	Customers leaving the store without purchasing all the items they need.
Affects	Shoppers.
The impact of which is	Customer will either, return to the store or purchase the item next time they visit the store.
A successful solution would be	Maintain a list of items the shopper needs and assist them with locating the items in the store.

The problem of	Customers leaving the store without purchasing all the items they need.
Affects	Store Owners.
The impact of which is	Shoppers will leave without purchasing everything on their shopping list.
A successful solution would be	Maintain a list of items the shopper needs so that the store may assist them with locating potentially forgotten items in the store.

The problem of	Customers leaving the store without realizing they have not purchased items that they need.
Affects	Store Owners.
The impact of which is	Shoppers will buy items somewhere else.
A successful solution would be	Monitor customer shopping habits and make recommendations while they are shopping in the store.

The problem of	Customers having to search for staff to provide assistance and wit around for that member of staff to come free.

Affects	Shoppers.
The impact of which is	Shoppers will leave the store without receiving assistance and as a result not purchase an item they needed.
A successful solution would be	Allow shoppers to communicate remotely with staff and continue with their shopping while they are waiting for a store assistant to come free..

1.3 Actors

Actors may be roles played by people, or they may be computer systems used by the people in order to assist with performing a business process.

1.3.1 Roles

Roles are the people actors (or stakeholders) that are impacted by the process. These are shown in Figure 1:

- Customer – Is anyone that enters the store who is not recognized as working in the store.
- Shopper – A customer who can be identified by the store as someone who shops here (often recognized by a store membership card in their name).
- Store Assistant – Provides assistance to the shopper.
- System Administrator – Looks after the computer systems and hardware used by the store.
- Cashier – Takes payment for items being purchased by the shopper.

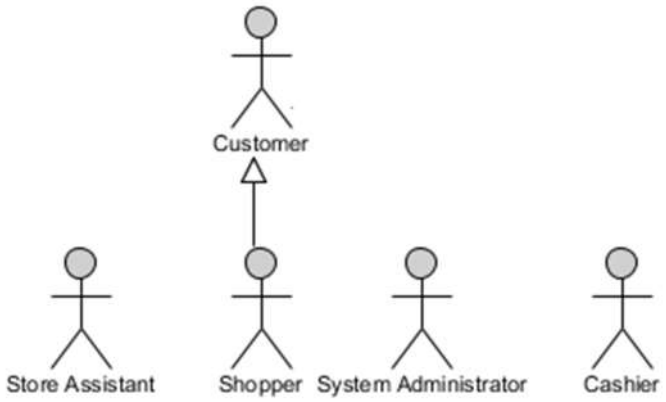

Figure 1: Role Actors

1.3.2 Systems

System actors are computer systems that assist the human actors (roles) with carrying out the process. The system actors in this example are shown in Figure 2:

- Customer Database – Contains a record of all customers registered with the store.
- Product Database – Is where a record of every item sold in the store is kept.
- Checkout – Takes payment from the customer at anytime upon request. It communicates with banking systems to verify the payment method and transfer funds, as appropriate.

- Item Detector – Detects tagged items leaving the store.
- Mobile Device – Reads any store related item with a barcode; this includes items for sale and the customer membership card. Maintains a list of items purchased their cost and location in the store. Informs the customer with messages and provides assistance upon request.

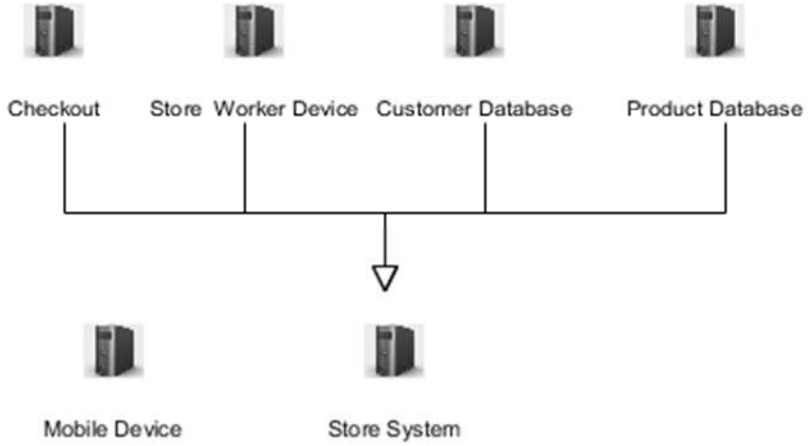

Figure 2: System Actors

- Store System – Any system owned by the store that may communicate with any other system, is given a generic name of Store System.

2 As-Is Business Model

The first model the business analyst traditionally builds is called the ‘as-is’ model. This captures the current business process and is used to compare the differences between the current process and the future process. Identifying changes and things that can stay the same gives an estimate the cost to build and return on investment of building a new business process.

2.1 Business Objects

These are the components currently in use in the shopping process. The systems used by the business today are shown in Figure 3:

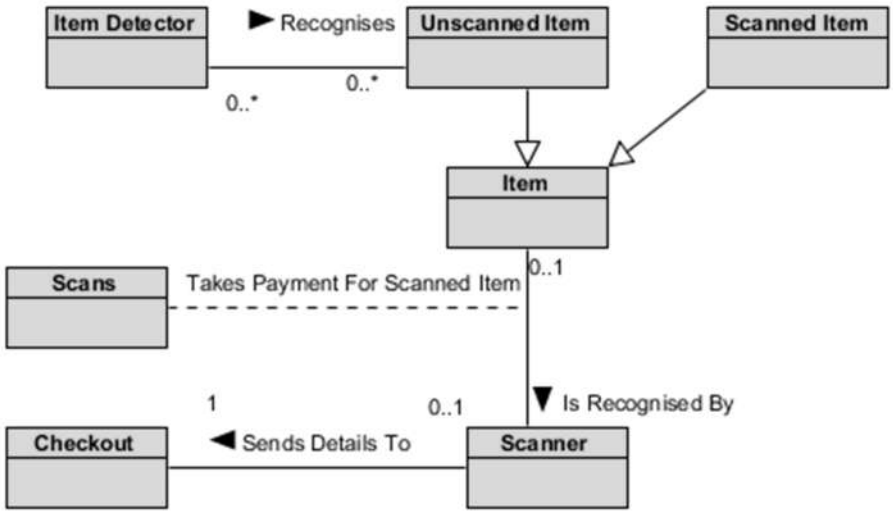

Figure 3: As-Is Business Objects Diagram

- The item detector recognizes unscanned items leaving the store. (Not strictly true; often it only recognizes the more expensive items.)
- Items are products that the store sells to its customers. An item may be scanned or unscanned. (It is assumed that scanned items will have any tags, recognized by the item detector, removed as they are scanned.)
- The scanner recognizes items as they are scanned.
- Scanning an item informs the checkout of details and cost to be added to the shoppers bill.
- The checkout takes customer payment in return for scanned items.

2.2 As Is Business Use Cases

A business use case describes a business process that gives benefit to an actor. The actor and the benefiting process are connected by a relationship.

The as-is business use cases describe 2 scenarios. These are shown in Figure 4:

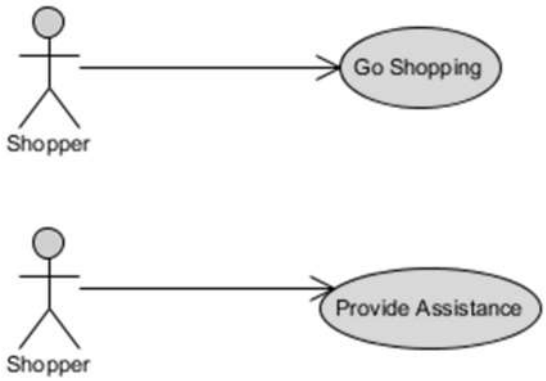

Figure 4: As-Is Business Use Case Diagram

- Go Shopping – This is the process by which the shopper purchases items in the store.
- Provide Assistance – The shopper is given help by store workers while they are shopping.

2.2.1 Go Shopping

The business use case process is described graphically by an activity diagram, as shown in Figure 5:

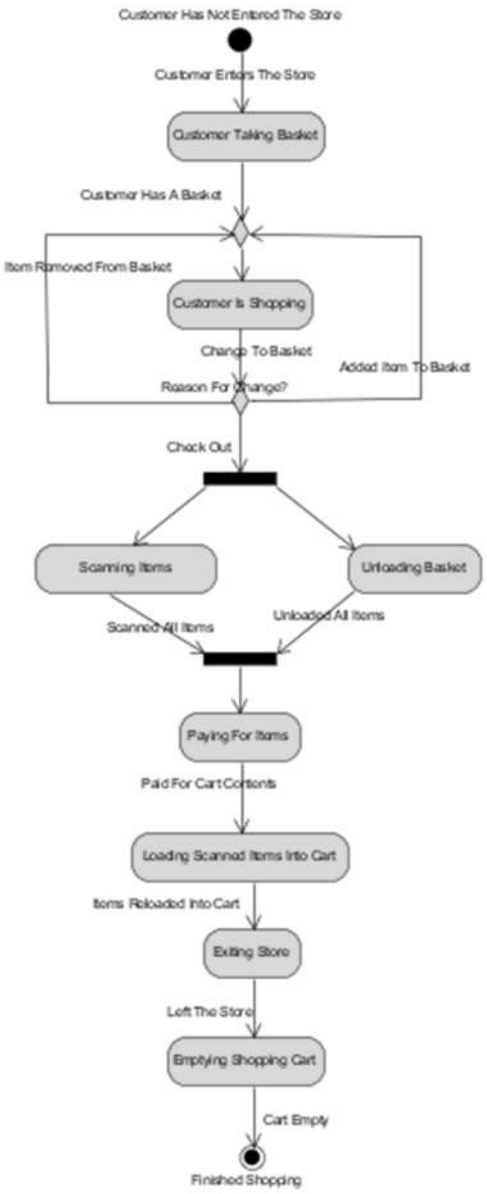

Figure 5: As-Is Buy Products Activity Diagram

Precondition: Before the process starts the customer is not recognized as being in the store.

Trigger: The customer takes a shopping basket (basket refers to any carrier used by the shopper to hold items while shopping in the store).

1. The customer takes a shopping basket while in the store, they are now a shopper.
2. As the shopper shops in the store they load up the basket with items. The shopper may add or remove items in the basket.
3. The shopper may checkout at anytime by unloading and scanning items in their basket.
4. Once scanning is complete the shopper pays for their scanned items.
5. The shopper then loads their scanned items back into the basket.

6. The shopper exits the store where any unpaid tagged items are detected.
7. The shopper empties their shopping basket and returns it to the store.
8. The use case ends.

Postcondition: The customer has completed their shopping.

Alternate Postcondition: Note that although not explicitly shown, there is an exception path whereby the shopper abandons their shopping basket at anytime during the above activity.

Additional information is included in Figure 5: in the form of the business objects that are used by the process. (The business objects are added for consistency checking purposes, and do not necessarily form part of the use case itself.)

2.2.2 Provide Assistance

Providing assistance in the As-Is model is a purely manual process. The activity is shown in Figure 6:

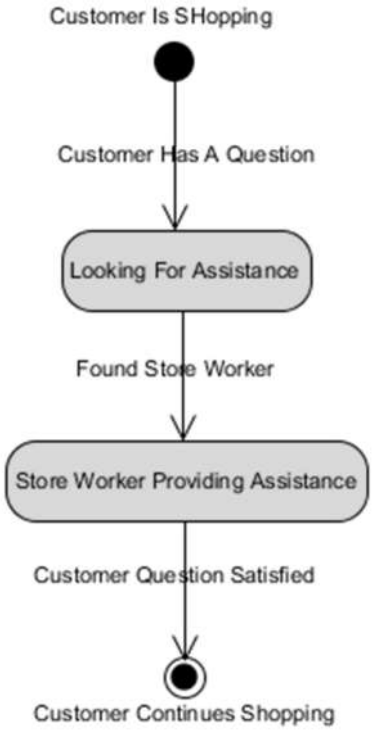

Figure 6: Provide Assistance Activity Diagram

Precondition: Customer is shopping in the store.

Trigger: The shopper has a question for a store assistant.

1. The shopper is looking for a store assistant.
2. The shopper gets the attention from a store assistant.
3. The store assistant is assisting the shopper with their question.
4. The shopper

Postcondition: The shopper continues with their shopping.

Alternate Postconditiom: Although not shown, it is possible that the shopper abandons their search for a store assistant and continues shopping with their question unanswered.

3 To-Be Business Model

In conjunction with the ‘as-is’ model, we build a ‘to be’ business model to describe the improved business process.

3.1 Business Objects

The business objects as they will become with the new process are described in Figure 7:

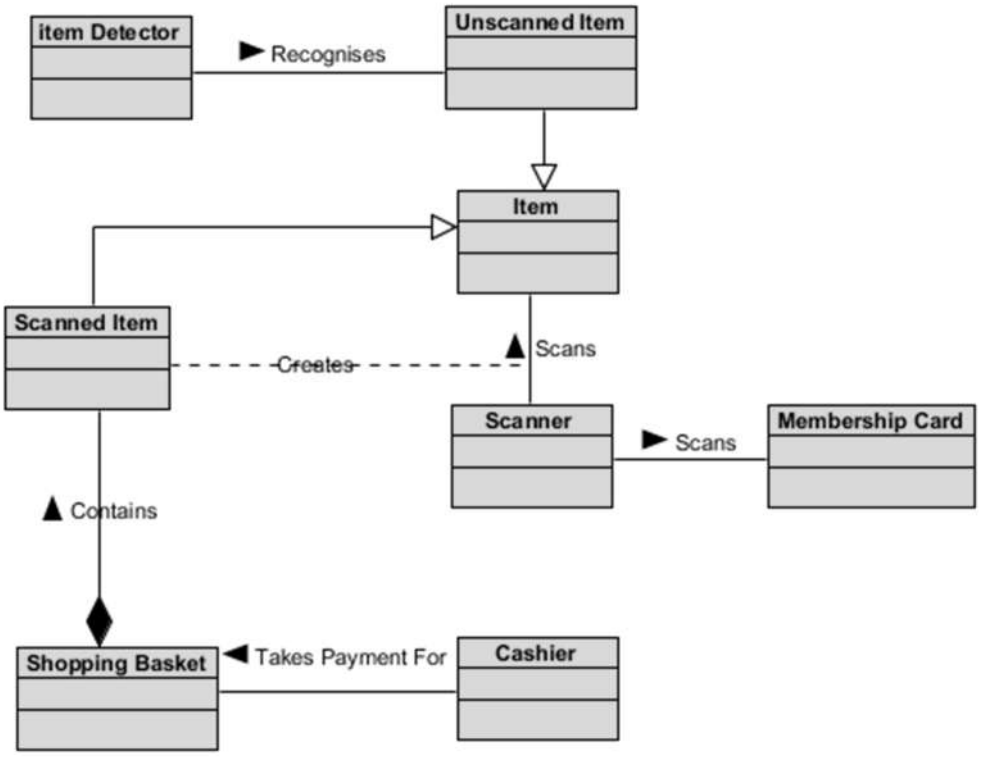

Figure 7: To-Be Business Objects Diagram

- The item detector recognizes unscanned items by either detecting tagged items. These are compared to items in the shopping basket.
- Items are classed as scanned items or unscanned items. An item is an instance of a product carried by the store.
- Items are scanned as they are entered into the shopping basket.
- The cashier reads the items in the shopping basket and takes payment for those items, producing a receipt for scanned items.
- The customer membership card is scanned in order to identify the shopper.

3.2 To-Be Business Use Cases

The to-be business use cases describe the same benefits to the same actors as the as-is process did.

These are shown in Figure 8:

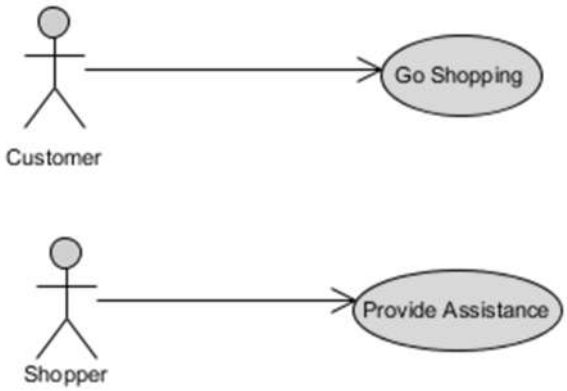

Figure 8: To-Be Business Use Case Diagram

3.2.1 Go Shopping

The future shopping activity shown in Figure 9:

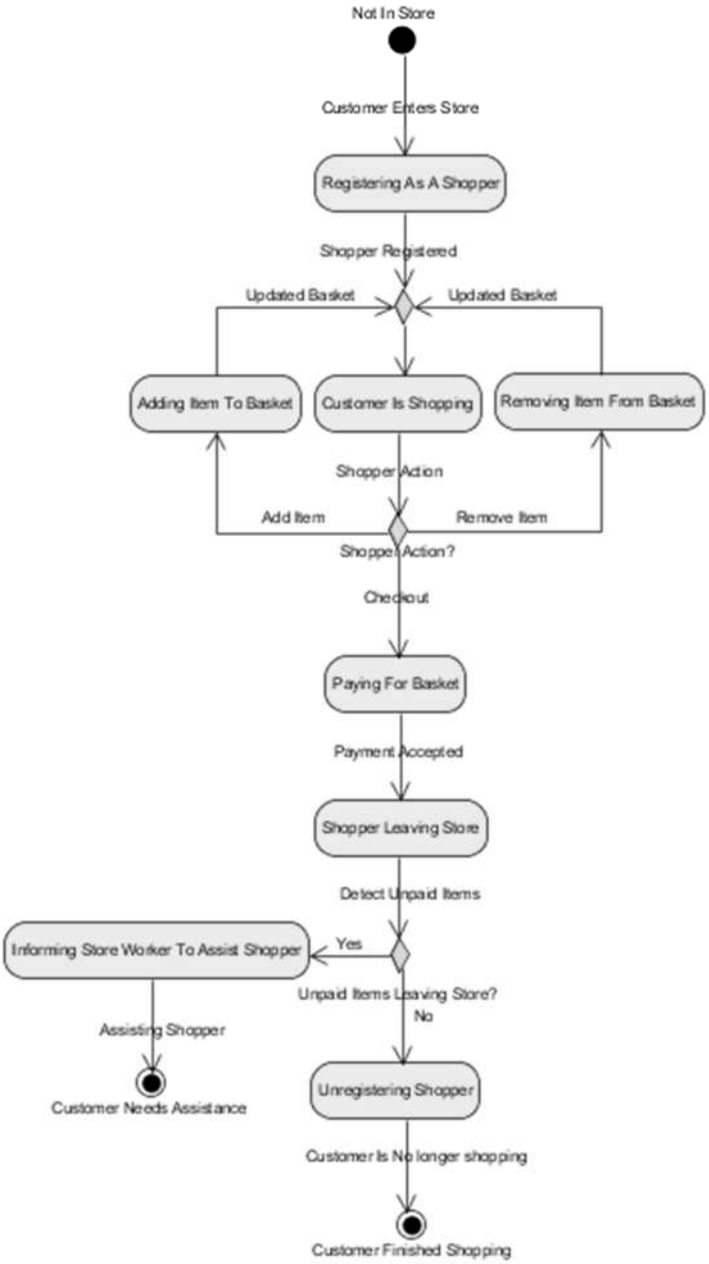

Figure 9: To-Be Buy Products Activity Diagram

The activity is similar to the as-is use case. Actions have been included for adding and removing items in the basket, since there will be a scanning action taking place.

When the shopper reaches the checkout, they no longer have to scan the items in the shopping cart. Nor does the shopper unload and reload their shopping basket at the checkout. The total cost of the items in the cart is known to the cashier without having to scan them.

As the shopper leaves the store, items in their basket are compared to those that were paid for.

How is this possible? By adding an intelligent scanning system to the shopping process. This system will track the cost of items being placed in the basket and also those being removed from the basket.

The shopper simply exits the store and empties their basket.

The mapping between the to-be business process and the systems that make this possible is shown in the next section.

3.3 Provide Assistance

The future store assistance activity shown in Figure 10:

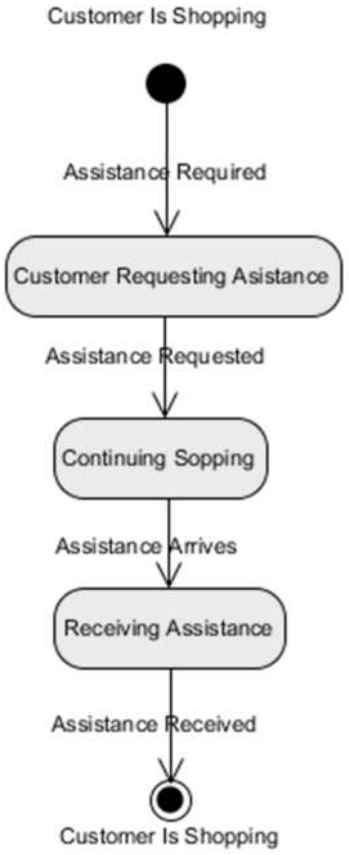

Figure 10: Provide Assistance Activity Diagram

The difference between the to-be activity and the assistance process described in the as-is use case, is that the shopper may continue shopping while they are waiting for assistance. The shopper no longer has to find a store assistant and wait for that person to become available. They simply send out a request for assistance and assistance arrives when the next store assistant becomes free.

4 Business To System Mapping

The business to system mapping model shows how the business process activities are mapped to computer systems that will automate the shopping process.

4.1 Business Architecture

The business architecture shows the systems that will be used to automate the shopping experience, and how they communicate with each other. Communication between systems is shown in Figure 11:

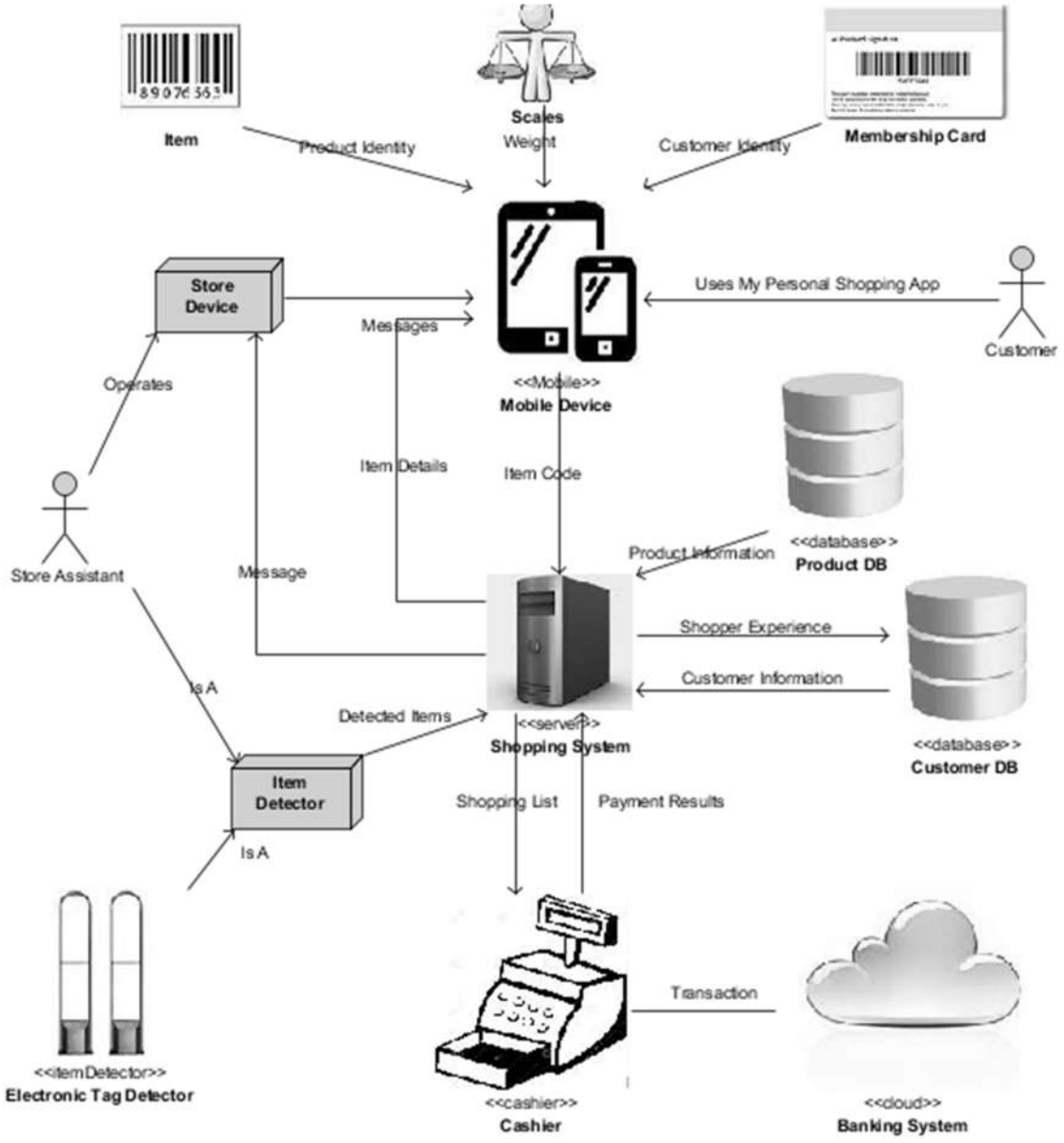

Figure 11: Shopping System Architecture Diagram

4.1.1 Computer systems

These nodes are computer systems that will require changes to functionality.

- The Shopping System is a new system that will be used to control the shopping experience. It communicates with the following nodes:
  - Mobile Device – sends messages to and receives scanned item codes from the Shopping System.
  - Cashier – communicates with the Shopping System by sending the item descriptions and cost in the shopping basket to, and receiving status about payment made by the shopper.
  - Item Detector – informs the Shopping System of items that pass by it. An item detector may be electronic, a manual inspection, or a combination of both.
  - Customer Database – Shopping System sends the customer Id to the Customer Database and is returned information about the customer.
  - Store Device – sends messages to and receives message from, the Shopping System.
  - Product Database – communicates information about store items to the Shopping System.
- A Mobile Device scans products added to and removed from the shopping basket, reads the customer store card and displays messages to the shopper. The mobile device may optionally communicate with intelligent scales in order to identify the cost of weighed items. A Mobile Device is any device capable of running the store shopping application, whether customer or store supplied.
- The product database contains all the information needed about products sold in the store. This is sent to the shopping system upon request.
- A Customer Database is used to track and validate shoppers. Also to provide customer information to the Shopping System.
- The Cashier system deals with making payment. It receives a total cost from the shopping system and ensures the correct payment is made by the shopper. The Cashier handles many forms of payment.
- The Store Device is used by employees to assist shoppers with their shopping experience.

4.1.2 Components

These nodes are equipment that send information to or receive information from computer systems.

- The Item Detector sends information about products leaving the store, to the Shopping System. An item detector may also be a store worker who manually confirms items leaving the store.
- Scales – are used to send information about weighed items to the Mobile Device.
- Membership Card – identifies the customer through the Shopping Application on the Mobile Device.

4.2 Business Activity To System Mapping

The actions in the ‘to-be’ business use case activity diagrams are mapped to systems, identified in the architecture diagram. The mapping from business activity to system architecture is shown in Figure 12:

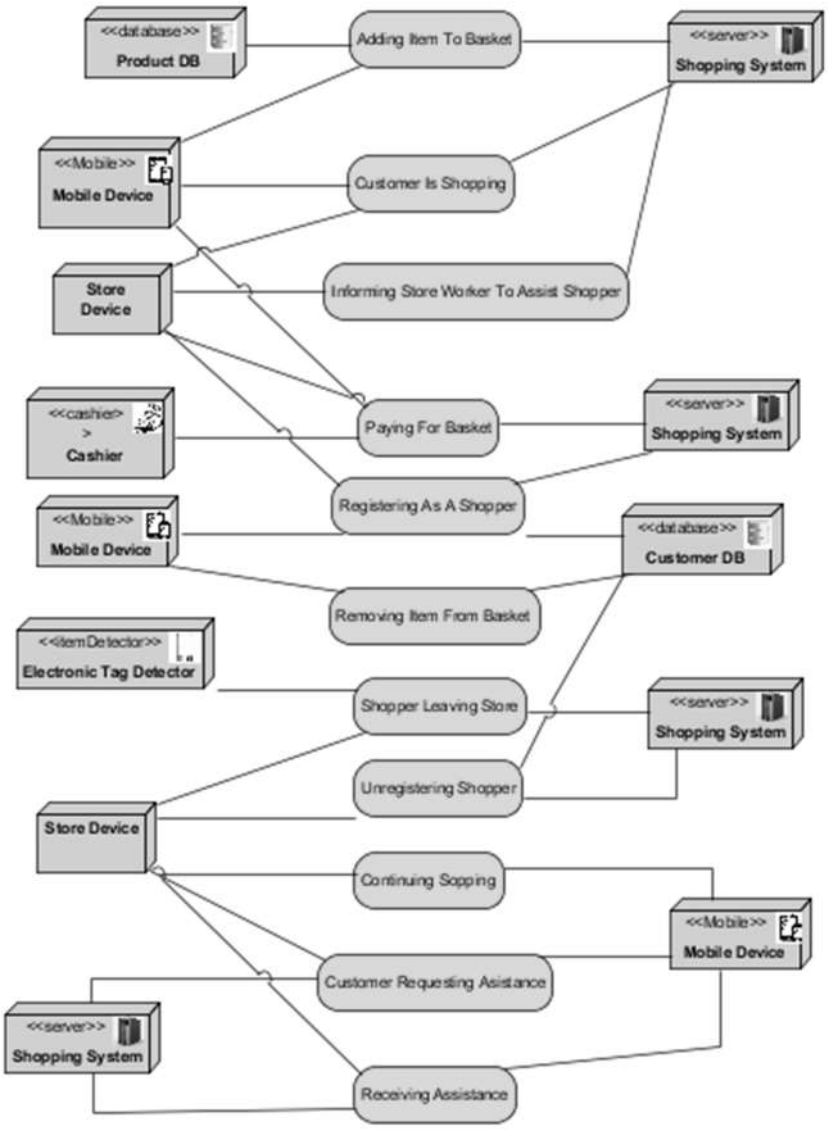

Figure 12: To-Be Business To System Mapping Diagram

Each business action, shown in yellow, is mapped to a system shown in blue. [Actions are listed alphabetically because that is how the tool orders them and it makes it easier to ensure that I have not missed any by following the tool order.]

- Every business action has a Shopping System mapping
- Every business action has a Mobile Device mapping.
- In addition to providing assistance to shoppers, the Store Device is mapped to any action which might cause a shopper status to change. The Store Device is also used with processes involving exiting the store.
- The Item Detector is involved with shoppers exiting the store.
- The Cashier system is involved with taking payment for items.
- The Customer Database is used when the customer registers and is updated with the customer status when the shopper exits the store.

- The Product Database is used when the shopper is adding items to their cart. (Removing an item does not need a product lookup since the item is already known by the mobile device.)

4.3 The Candidate System Use Case Model

With the business actions mapped to enterprise systems, the last part of the business model is to identify potential system use cases for each system that will satisfy the business activities.

Where a business action is assigned to a system, indicated by the relationship lines in Figure 12: we now create potential system use cases to model those relationships. (Potential because each candidate use case shown in Figure 13: may take multiple instances, complex use cases may be split into multiple system use cases, and some may be combined.) Remember that a system use case may only capture activity for a single system, so where a business action connects to multiple systems, there will be at least 1 system use case for each system.

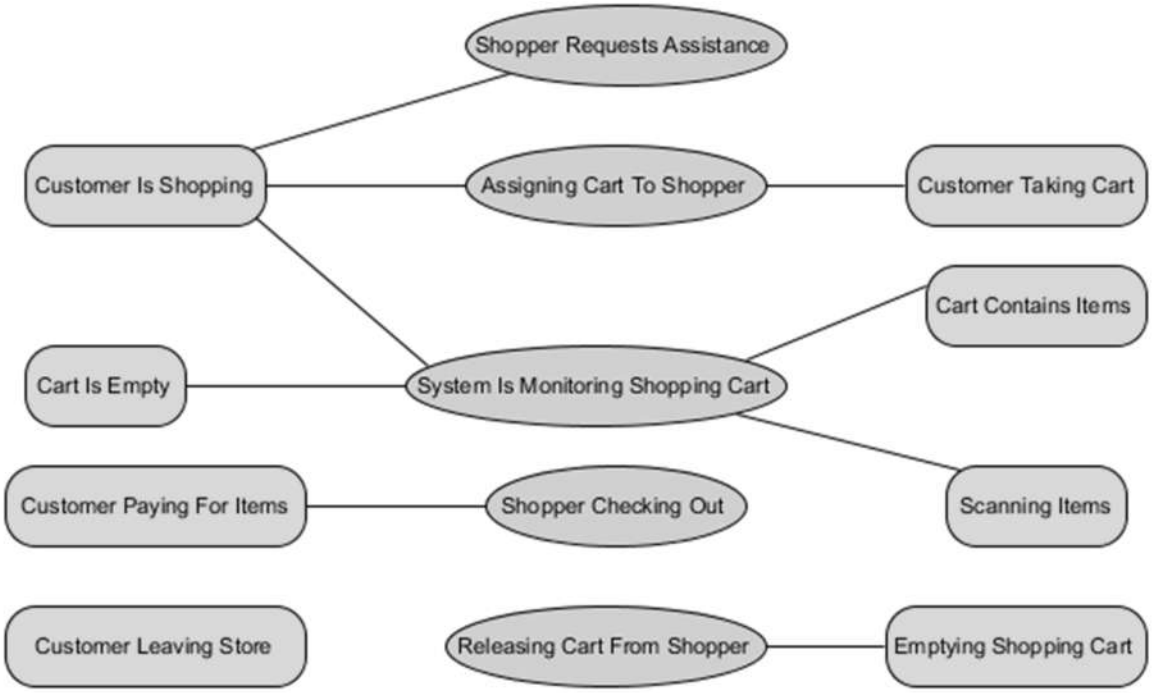

Figure 13: Business Action Mapping To Store System Candidate Use Cases

Figure 14: Business Action Mapping To Store System Candidate Use Cases

4.4 Systems

Each system in the architecture has its own set of use cases. The system with which we are mostly concerned is the Shopping Cart System, however, changes are needed to the Customer Database, Product Database and the Cashier systems.

The Item Detector detects store items as they pass through it. The context diagram for the Shopping Cart System is shown in Figure 15:

Figure 15: Shopping Cart System Context Diagram

4.4.1 Shopping Cart Use Cases

The following functions are performed by the shopping cart system. The use cases capturing these functions are shown in Figure 16:

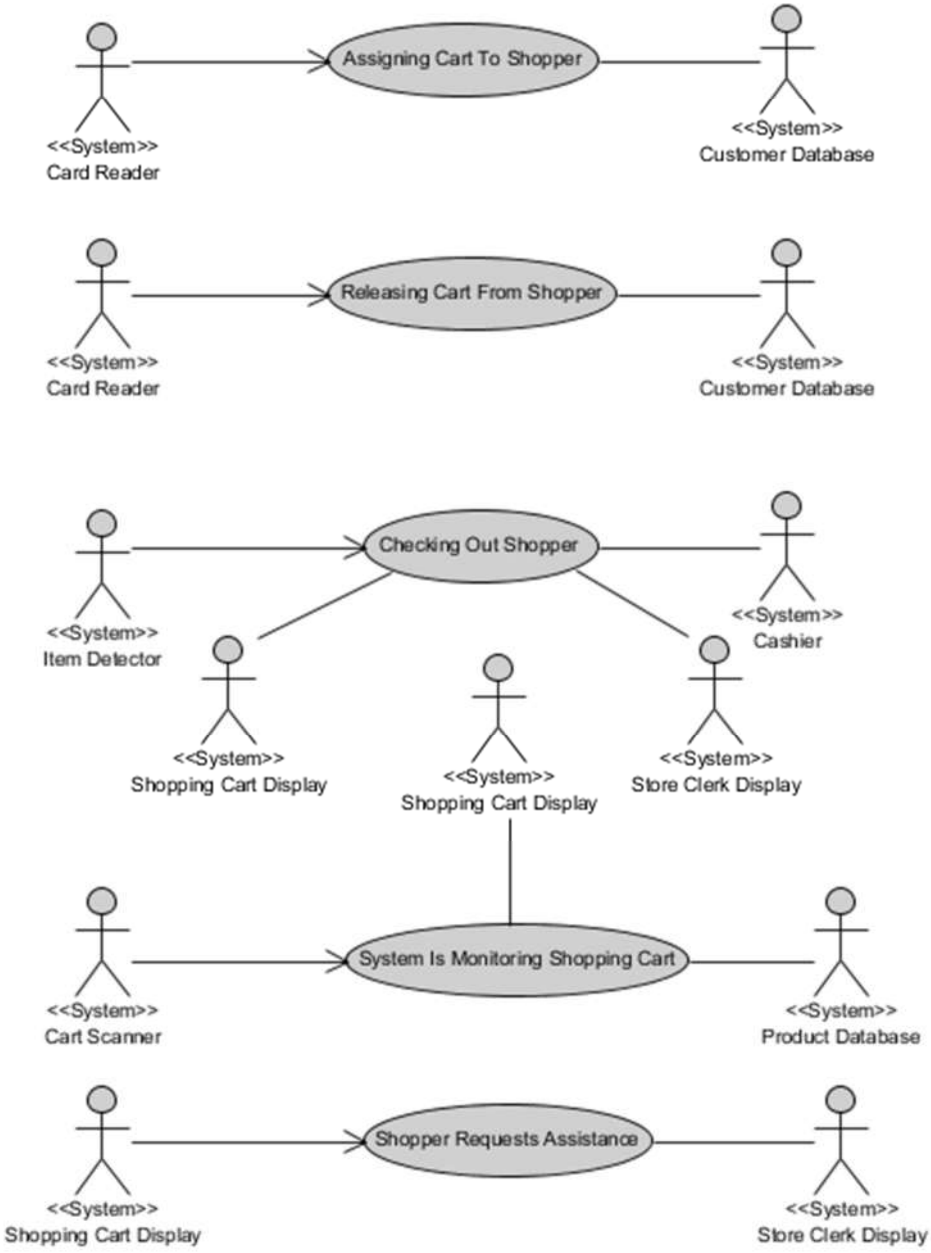

Figure 16: Shopping Cart System Use Cases

4.4.1.1 Assigning Cart To Shopper

Shoppers are identified by their store card, which includes a unique shopperId attribute. The customer card is read by the cart (bank cards are a future option for investigation), and the customer is identified by the customer database. The customer may be refused to use the shopping system for many reasons (in which case they will shop using the current as-is process). Otherwise the customer is approved as a shopper and the new shopping experience begins.

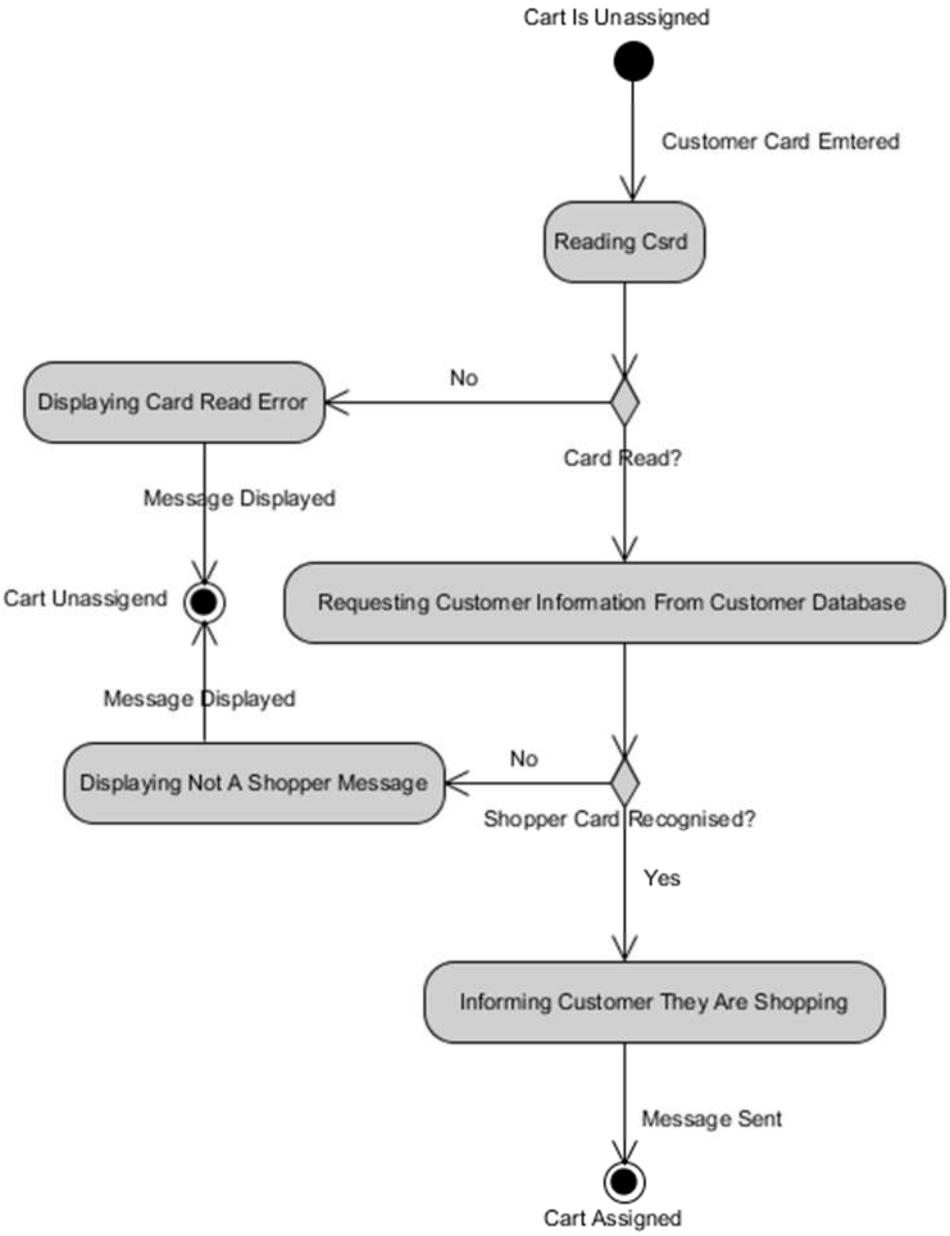

Figure 17: Assigning Cart To Shopper Activity Diagram

Precondition:

The shopping cart is not assigned to a shopper

Trigger:

A customer enters their store card and the cart reads their Shopper Id

Steps:

1. Shopping cart card reader reads the card
2. Card is able to be read and the shopping cart sends the customer id to the customer database and requests the shopper information

3. The customer database responds with a valid shopper information
4. The cart informs the shopper that they have been accepted and they may start shopping
5. The use case ends

Postcondition

Customer Is A Shopper.

Alternative Path:

None.

Exception Path:

E.1. Card Cannot Be Read

6. At step 2, the customer card cannot be read
7. The customer is displayed a card read error
8. The use case ends

Postcondition:

The customer is not allowed to use the shopping system.

E.2. Customer Is Not Allowed To Use The Shopping System (or they are already assigned to a shopping cart)

9. At step 3, the customer database informs the shopping cart system that the customer is not a shopper or that they are already assigned to a cart
10. The customer is displayed a message to see membership
11. The use case ends

Postcondition:

The customer is not shopping.

4.4.1.2 Releasing Cart From Shopper

A customer is assigned to a single cart when their store card is inserted into the card reader. If the card is removed the customer is no longer assigned to the cart. If there are unpaid items in the cart the shopper is informed that they have not finished shopping. (The store clerk could also be informed of carts containing unpaid items. This may allow for easier cleanup within the store.)

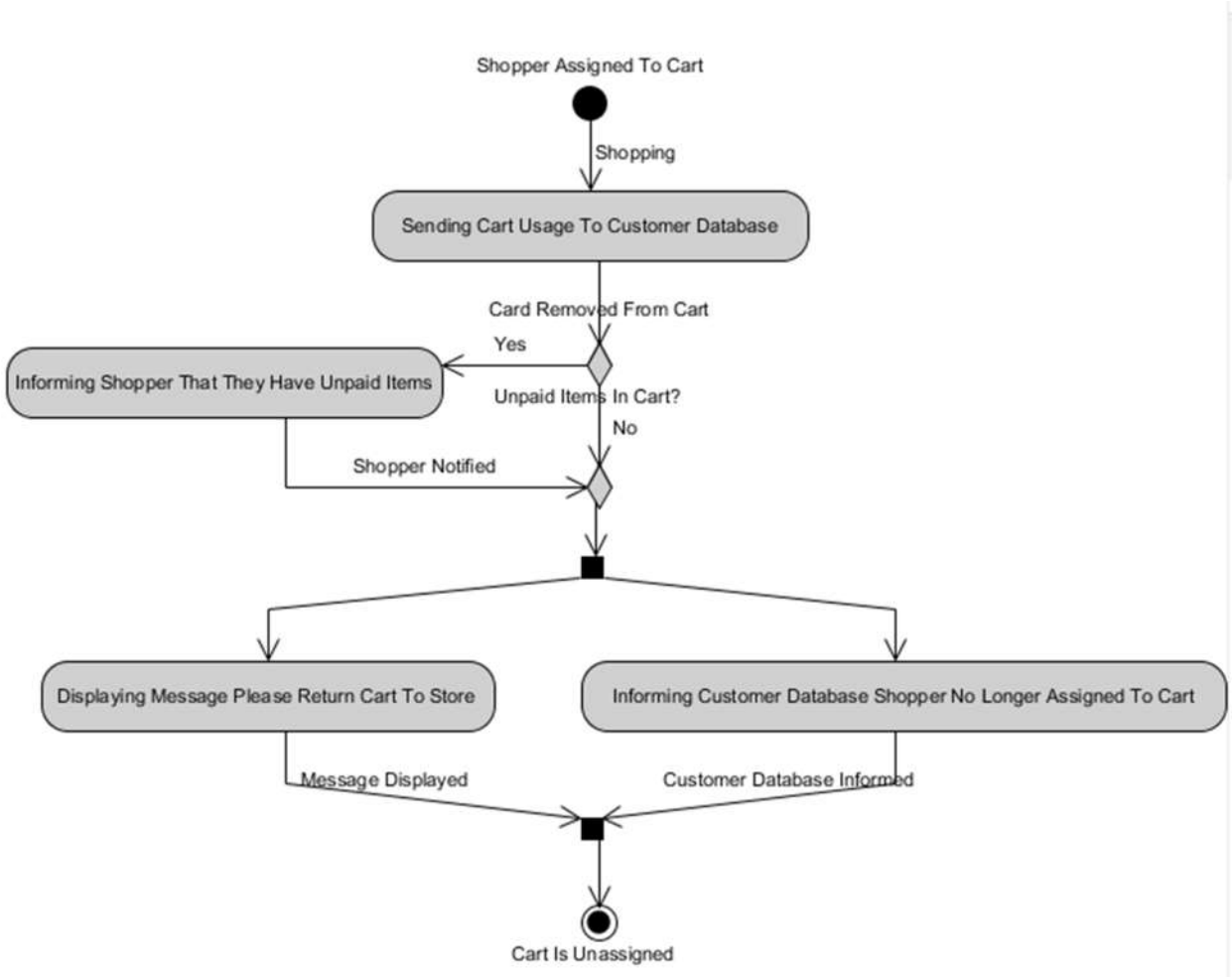

Figure 18: Releasing Cart From Shopper Activity Diagram

Precondition:

The shopping cart is assigned to a shopper

Trigger:

The customer is shopping

Steps:

1. The shopping cart monitors the card reader
2. The card is removed from the cart
3. There are no items in the cart and

4. The shopping cart informs the user to return the cart to the store	5. The shopping cart informs the customer database that the customer is no longer shopping

6. Use case ends

Postcondition

The shopping cart is available to be used by another customer for shopping.

Alternative Path:

A.1 Cart Is Not Empty

7. At step 3, there are items in the cart and the shopper is informed that they have unpaid for items in the cart
8. Use case continues from step 3

Exception Path:

None.

4.4.1.3 System Is Monitoring Shopper

The shopping cart system will monitor store items as they are added to or removed from the shopping cart. A total cost (and optional list of items in the cart) is displayed to the shopper at all times.

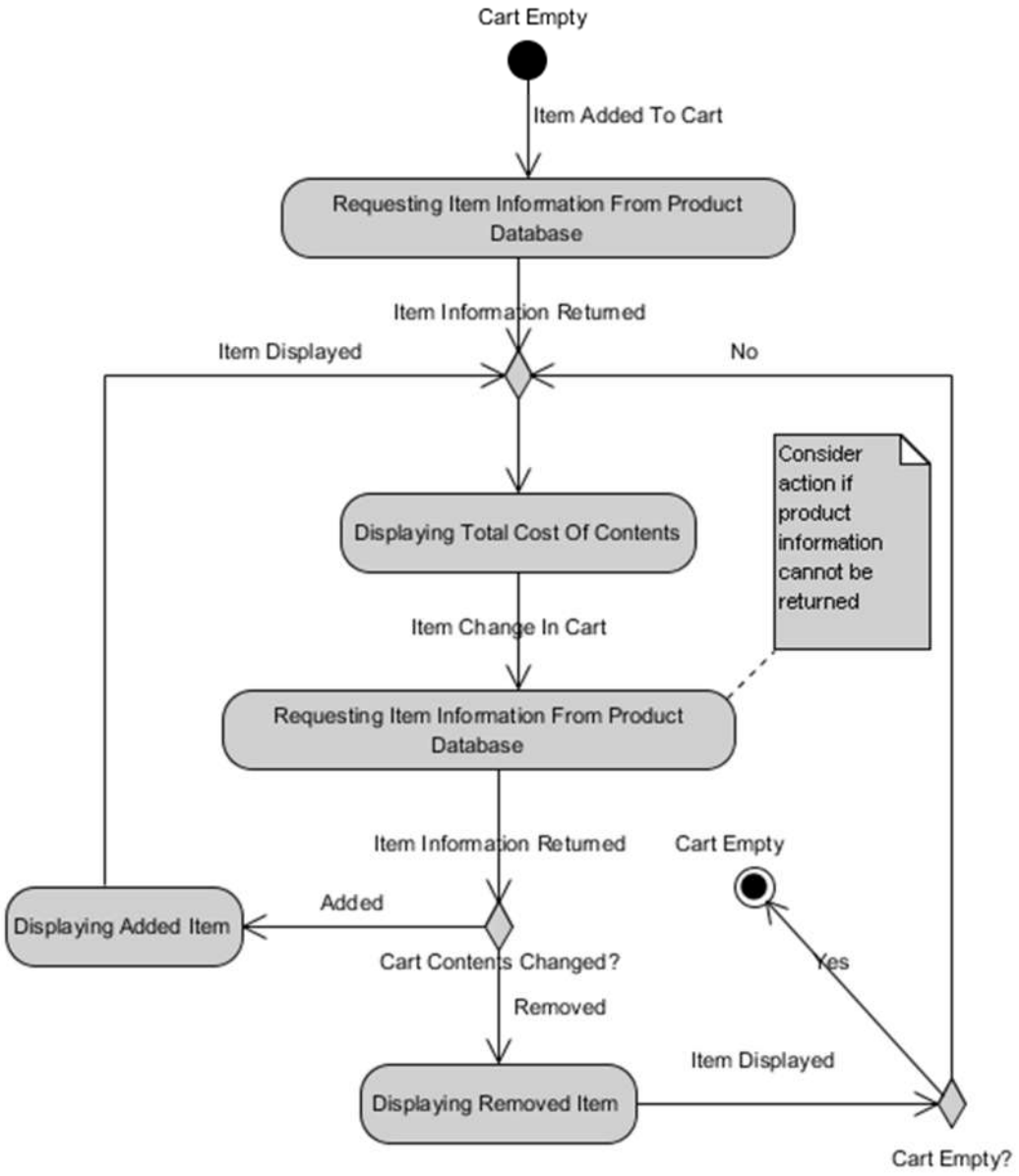

Figure 19: System Is Monitoring Shopper Activity Diagram

Precondition:

The shopping cart is empty (this use case functions whether or not the cart is assigned to a shopper)

Trigger:

A store item is recognized as being added to the shopping cart

Steps:

1. The system requests the product information from the product database
2. The product database returns cost of the item and product information
3. The system is displaying the total cost of items already in the shopping cart

4. The contents of the cart are changed
5. The system requests the changed product information from the product database
6. The product database returns the product information and cost of the item (may need to add consideration for what happens if the product is not found)
7. An item was removed from the cart and the system updates the display to show that the item has been removed
8. The cart is empty and the use case ends.

Postcondition:

The cart is empty.

Alternative Path:

A.2 Item Is Added To The Cart

1. At step 7, an item was added to the cart and the system updates the display to show that the item was added
2. The use case continues from step 3

A.3 Cart Is Not Empty

3. At step 8, the cart has items in it and the use case continues from step 3

Exception Path:

None.

4.4.1.4 Shopper Requests Assistance

At any time a shopper is assigned to a cart, they may request assistance from a store worker.

Figure 20: Shopper Requests Assistance Activity Diagram

Precondition:

Shopper is assigned to a cart

Trigger:

Shopper requests assistance

Steps:

1. System sends a request to the store clerk display from the cart and shopper id
2. The use case ends.

Postcondition

Assistance message is sent to the store clerk display.

Alternative Path:

None.

Exception Path:

None.

4.4.1.5 Shopper Checking Out

Once the shopper reaches the checkout area with a cart that is assigned to them, they will be automatically billed and requested to make payment.

If the cart is not assigned to a shopper or if payment was not completed, a store clerk is informed to assist the shopper.

Upon completion of payment, a check is made to verify that the items exiting the checkout are equivalent to the items that were paid for. If there is a discrepancy a store clerk is informed to assist the shopper.

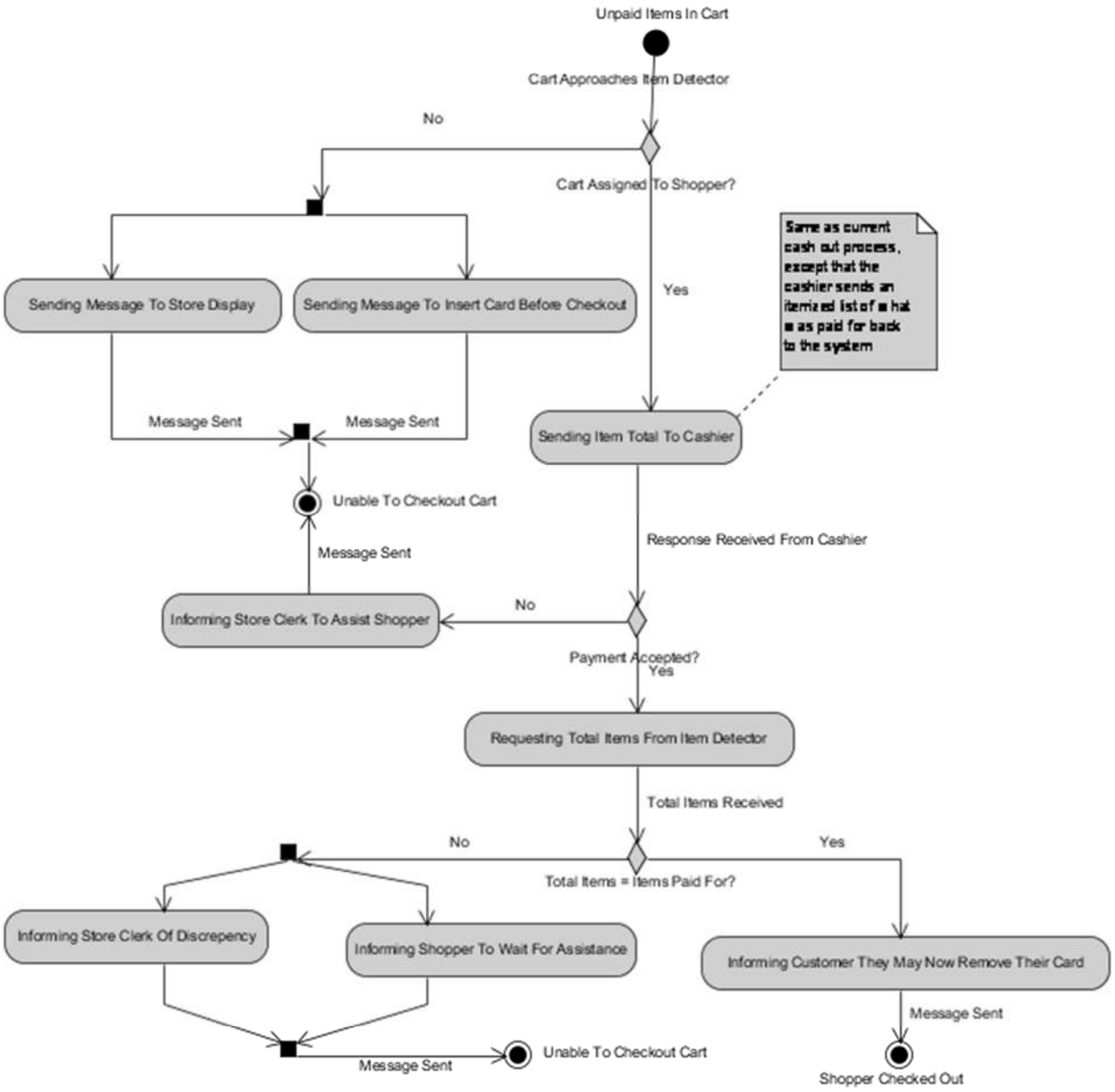

Figure 21: Shopper Checking Out Activity Diagram

Precondition:

There are unpaid items in the shopping cart

Trigger:

The cart reaches the checkout

Steps:

1. The cart is assigned to a shopper and the total amount for items in the cart is sent to the cashier system
2. The cashier system reports that payment has been accepted for the items
3. The system requests the total for items passing through the item detector

4. The item detector returns information about the items in the cart (could simply be the number of items or it could return an identifier for each item)
5. The total of all items is equal to the items paid for and the shopping cart informs the shopper that they may now remove their card and exit the store
6. The use case ends.

Postcondition

The shopper has checked out their items from the store.

Alternative Path:

None.

Exception Path:

E.1. Cart Is Not Assigned To A Shopper

7. At step 1, the cart is not assigned to a shopper

8. The system informs the store clerk that the cart cannot be checked out without a shopper assigned	9. The system informs the customer that they need to insert their shopping card in order to checkout

10. The use case ends.

Postcondition

The shopping cart was not checked out.

E.2. Payment Is Not Accepted

11. At step 2, the cashier system informs the cart that payment was not accepted
12. The system informs the store clerk to assist the shopper with checking out their cart
13. The use case ends.

Postcondition:

The shopping cart was not checked out.

E.3. Items Not Paid For

14. At step 5, there is a discrepancy between items paid for and items exiting the store

15. The system informs the store clerk of the discrepancy between items leaving the store	16. The system informs the shopper to wait for assistance

17. The use case ends

Postcondition:

The shopping cart was not checked out.

4.4.2 Other Systems

In addition to the 'new' Shopping Cart system, several other existing systems are impacted by the business needs.

4.4.2.1 Item Detector

The following additional function is performed by the Item Detector system. The use case capturing this function is shown in Figure 22:

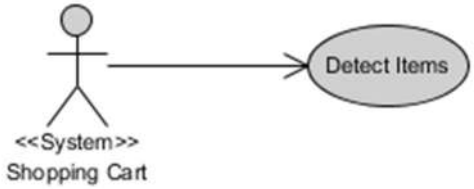

Figure 22: Item Detector System Use Cases

- As the shopping cart reaches the checkout the item detector reads the items in possession of the shopper and sends this information to the shopping cart system upon request.

4.4.2.2 Product Database

The following functions are performed by the Product database system to support the shopping cart system. The use cases capturing these functions are shown in Figure 23:

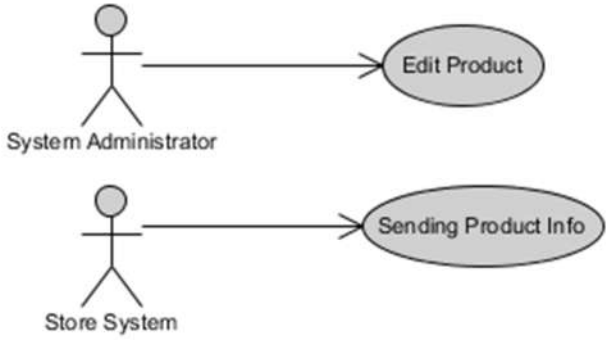

Figure 23: Product Database Use Cases

- The system administrator has the ability to add, remove and change products in the product database. (Existing function.)
- Upon request the product database will send product information to a requesting store system.

4.4.2.3 Customer Database

The following functions are performed by the Customer Database system to support the shopping cart system. The use cases capturing these functions are shown in Figure 24:

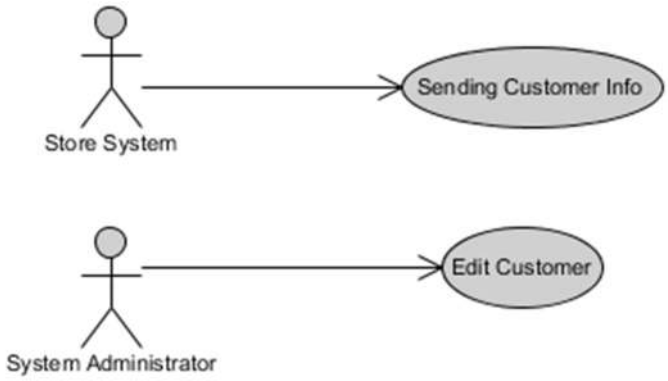

Figure 24: Customer Database System Use Cases

- The system administrator has the ability to add, remove and change shopper information in the customer database. (Existing function.)
- Upon request the customer database will send shopper information to a requesting store system.

4.4.2.4 Cashier

The following functions are performed by the Cashier system. In order to support the shopping cart system. The use cases capturing these functions are shown in Figure 25:

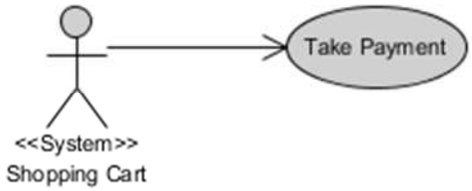

Figure 25: Cahier System Use Cases

- The cashier system receives item details from the shopping cart system and returns a payment status.

4.5 Summary

The use case model captures the functionality to be implemented by the project team. Use cases are structured in such a manner that each can be implemented independent of the others in the project. Note that each us case is assigned to 1 and only 1 system; as such implementing a single use case may not achieve any of the project objectives. In order to determine which order to implement the use cases, we need to trace them to the business model, as shown in Figure 26:

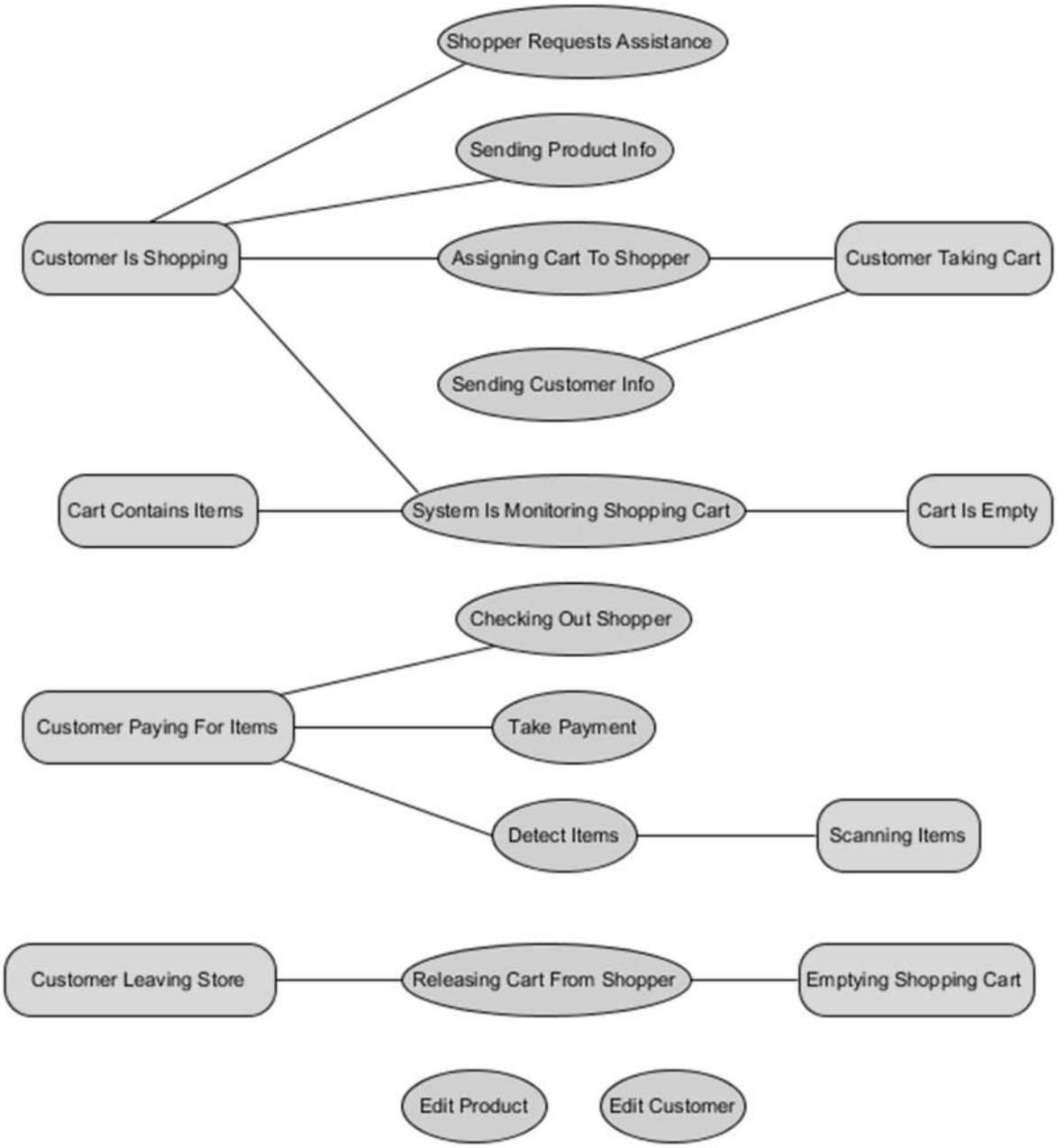

Figure 26: Business Activity To System Use Case Mapping

The Customer Is Shopping business activity is implemented by Shopping Cart system use cases, Shopper Requests Assistance, Assigning Cart To Shopper and System Is Monitoring Shopping Cart, also by the Product Database use case, Sending Product Info.

Customer Taking Cart business activity is implemented by the shopping cart system use case Assigning Cart To Shopper and the Customer Database system use case, Sending Customer Info.

Cart Contains Items business activity and Cart Is Empty business activities are satisfied by the System Monitoring Shopping Cart system use case.

The Customer Paying For Items business activity is satisfied by the implementation of systems use cases, Take Payment, Checking Out Shopper and Detect Items, which are assigned to the shopping cart, cashier and item detector systems resp.

Scanning Items business activity is satisfied by the item detector system use case Detect Items, which detects items leaving the store.

The Customer Leaving store business activity is partially satisfied by the Releasing Cart From Shopper use case.

The Emptying Shopping Cart business activity is partially satisfied by the implementation of the systems use case Releasing Cart From Shopper. (The process of removing items from the cart is performed manually by the shopper.)

The Edit Customer and Edit Product system use cases, are derived from decisions made about the architecture, and have no direct relationship to the business use cases. (Similar use cases can be derived for maintenance of the shopping cart, the scanner and the cashier systems.)

Figure 26: is a traceability diagram used to demonstrate the requirements from business needs to system implementation.

5 Extensions

Technology used is not described and will almost certainly have some impact on the analysis model. How does one make an intelligent shopping cart that is cheap enough to supply to many shoppers such that the costs do not outweigh the savings made by the system?

5.1 Meeting Objectives

How well does the to-be system model meet the business objectives by solving the problem statements?

The goals are stated in terms of Problem Statements which are derived from the project vision.

- The problem of it takes too long to check out.

Time to checkout is reduced because:

    - The cart does not need to be unloaded at the checkout
    - The cart does not need to be reloaded at the checkout
    - Items do not need to be scanned at the checkout.
- The problem of loading and unloading the cart too many times.

The shopping cart is loaded once and unloaded once. The process of unloading and reloading the cart at checkout is removed.

- The problem of the shopper does not know how much a cart of items is going to cost until checkout occurs.

The shopper is continuously updated with the cost and number of items in their cart.

5.2 Options

The shopping cart system could potentially be distributed over every shopping cart in the store. Each cart tracks its own items and the shopping cart system controls the process dependent upon the status of each cart.

When I first envisioned the shopping cart, I imagined each item having a (magnetic) strip that identified the product when it was inside the cart. The questions arise as to what constitutes ‘inside’ the cart and is it feasible to tag every store item with an electronically detectable mark?

An alternative is to have the shopper scan every item as they add to or remove from the cart. This may cause too many inconsistencies between what is in the cart and what was scanned.

Loading the cart - manually scanning items as they enter and leave the cart; how do we know that an item was added or removed without scanning? Traditional scanners today use weight to identify added or removed items. This is probably not feasible with a shopping cart. Another option is to simply count the number of items in the cart. If the number changes inform the shopper that they need to scan the added or removed item. Keep the customer informed of the number of items in the cart for their own personal verification.

However the cart items are tracked, the important piece of equipment is the Item Detector at the checkout. This will inform the store clerk during Checkout, if there are any inconsistencies between what is being paid for and what is being removed from the store.

The cart should be able to display a list of items in the cart and have the shopper be able to verify this with what they expect to be in the cart.

At anytime the shopper may perform a ‘reset’ of the shopping cart by rescanning all items in the cart in order to verify what is being recorded by the cart.

6 Credits

Visual Paradigm – used to draw diagrams, model the business and systems analysis.

Microsoft Office – Used for publishing this article.

Analysis Through Pictures – lulu.com