**Software Requirements Specification**

**for**

**Self Checkout Machine**

Version 1

Prepared by Group7: Eddie Torres, Felix Lang, Huda Ali, Ella Brink

CIS453

October 15th, 2023

1 Introduction

1.1 Purpose

This Software Requirements Specification(SRS) will provide a detailed description of a Self Checkout Machine that will streamline contactless payment, keep track of inventory, and provide a useful tool to a store and its employees. This SRS also provides the developers with a framework to ensure a successful implementation of the system.

1.2 Scope

This document specifies the requirements for the following capabilities:  
 1. Item scanning - Adding/Removing from cart

2. Purchasing Process (Cash and Card)

3. Additional Store Features (Coupons, Loyalty Cards)

4. Inventory Tracking

1.3 Overview

This document follows the recommended format specified in the IEEE Std

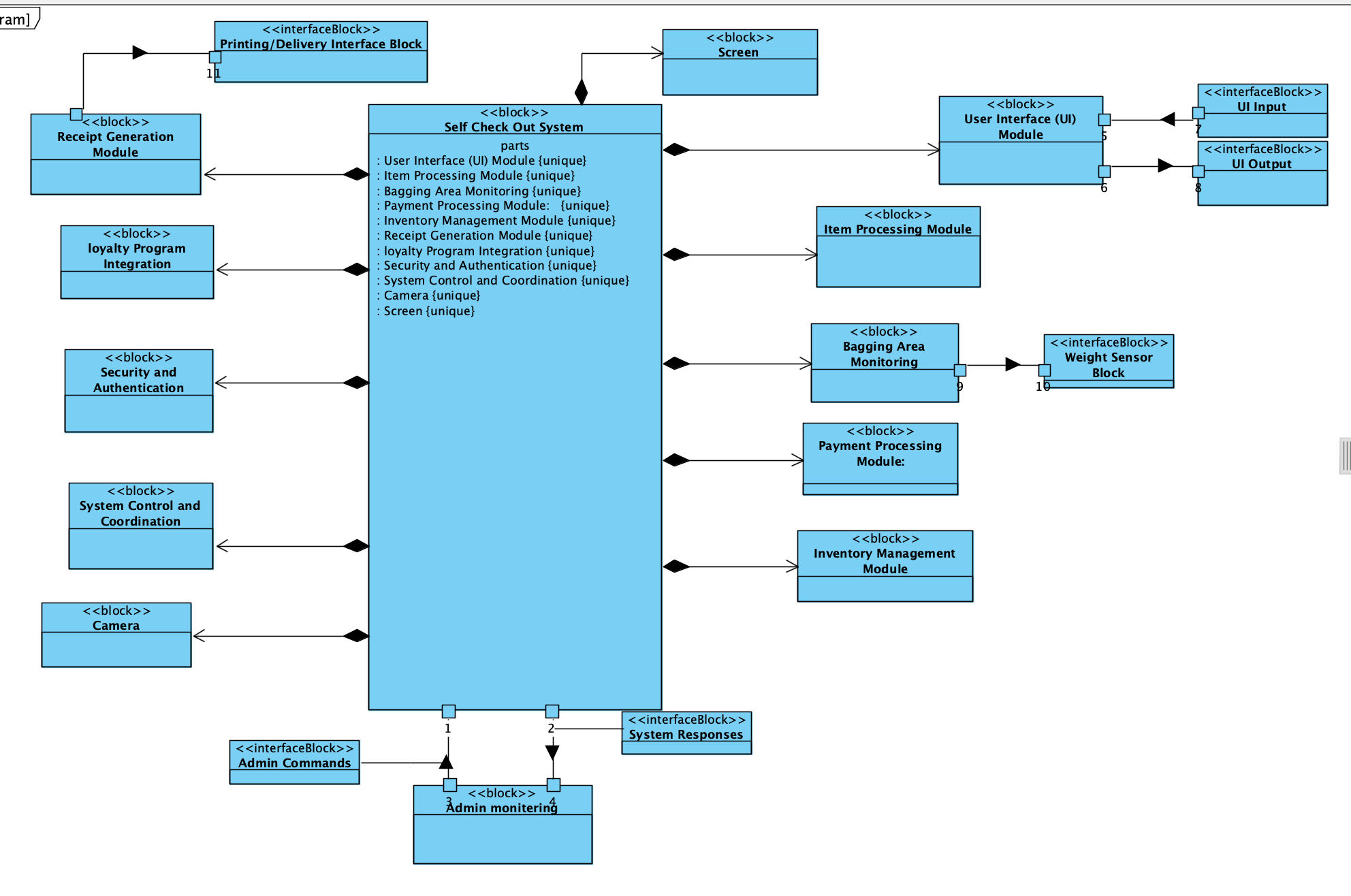
830 -1998 Recommended Practice for Software Specifications. For Section 3,

The specified template A.5 for organizing information by feature is followed.

2 Overall Description

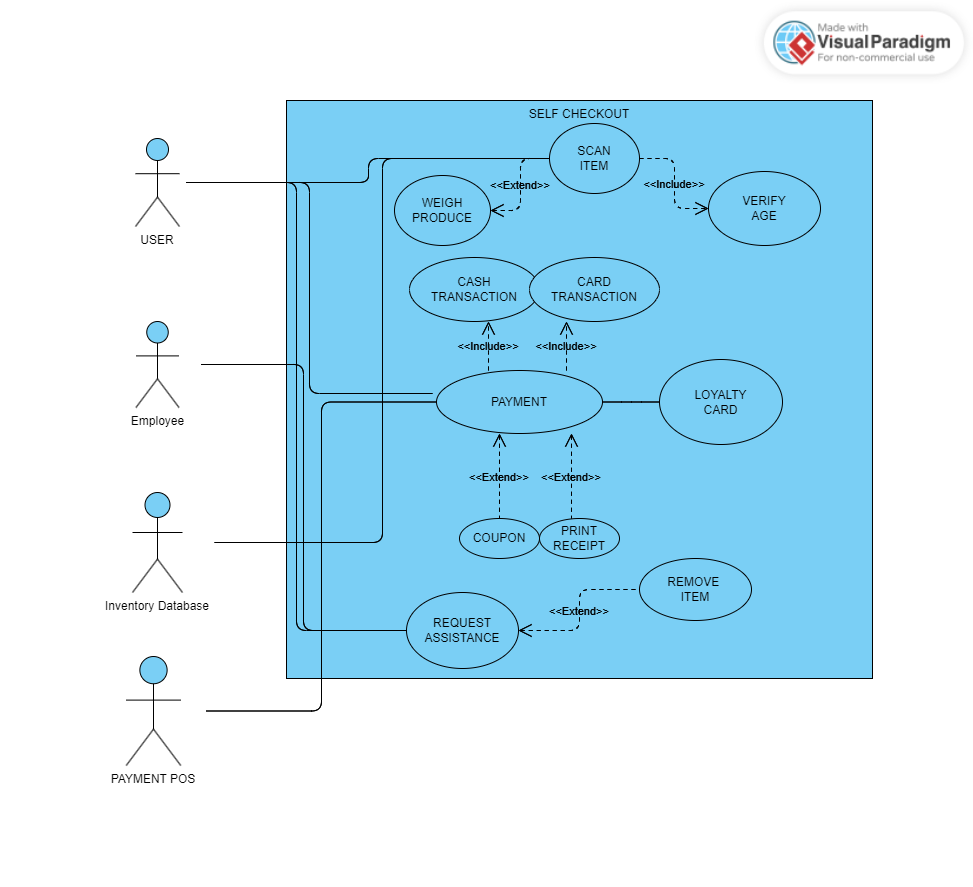
2.1 Product Perspective   
 This Self Checkout Machine is intended to provide a seamless shopping   
 experience while maintaining independence during the process. The

benefits of independence during shopping include a faster sales process, while avoiding invading the customers privacy, and simplifying the process for said customers. This will be done by taking advantage of the machine and allowing the responsibility to partially fall off of the employees.



**Figure 1 System Block Diagram**

2.2 Product Functions

The following Use Case Diagram depicts the users of the system and the intended way in which they interact with the system.

**Figure 2 Self Checkout Machine Use Cases**

2.3 **Use Case Descriptions**

Use Case: Weigh Produce

| General Characteristics | Weigh produce and use mass to calculate price for produce to be charged to the customer. |
| --- | --- |
| Intent | Allow the customer to measure the amount they took on their own |
| Scope | Must be able to calculate price per mass of all vegetables sold by store |
| Primary Actor | User |
| Secondary Actors | Inventory Database. |
| Preconditions | Must be marked as the correct product to calculate the price correctly. If the wrong price per mass is used, the wrong price will be calculated. |
| Assumptions | Must be able to accept all produce sold and marked as had by inventory. |
| Trigger | Successful Completion of Use Case: Produce price is calculated for customer by the scale and machine. |
| Success Post Conditions | Customer is given correct price for their produce |
| Failed Post Conditions | Customers are either given incorrect prices for their produce by wrong produce input or the scale causes inaccuracy. |

Use Case: Scan Item

| General Characteristics | Allow the User to scan their own items as they checkout |
| --- | --- |
| Intent | Provide the customer with an independent shopping experience |
| Scope | Must be able to scan all items that are marked as had in the inventory database. |
| Primary Actor | User |
| Secondary Actors | Inventory Database |
| Preconditions | User has selected items to scan and brought them to the machine |
| Assumptions | The inventory database may be incorrect which may cause conflict for the machine. |
| Trigger | Successful Completion of Use Case: User’s item is scanned and added to their purchase list |
| Success Post Conditions | Item is scanned and added to list to purchase |
| Failed Post Conditions | Item is not scanned or not added to the list of items to purchase |

Use Case: Verify Age

| General Characteristics | For certain items like Alcohol that require a minimum age to drink. |
| --- | --- |
| Intent | Ensure no rules are broken and to incur zero liability |
| Scope | Must be able to scan IDs from every state. |
| Primary Actor | User |
| Secondary Actors | Employee |
| Preconditions | Customer is of legal age for the item with the minimum age. |
| Assumptions | Customers not from the States may need to use a regular checkout with a cashier. |
| Trigger | Successful Completion of Use Case: Customer’s age is verified and they are sold or not sold the item based on their age. |
| Success Post Conditions | Customers age is verified |
| Failed Post Conditions | Customers age is not verified |

Use Case: Payment

| General Characteristics | Allow the Customer to purchase their items |
| --- | --- |
| Intent | Main Process of a Store |
| Scope | Must accept cash and card |
| Primary Actor | User |
| Secondary Actors | Payment POS / Inventory Database |
| Preconditions | Customer must have enough money |
| Assumptions | Customer might use cash or card |
| Trigger | Successful Completion of Use Case: Customer is able to purchase their items |
| Success Post Conditions | Customer purchases their items |
| Failed Post Conditions | Customer cant purchase their items |

Use Case: Cash Transaction

| General Characteristics | Allow Several Forms of Currency |
| --- | --- |
| Intent | Allow Several Forms of Currency |
| Scope | Accepted Currency |
| Primary Actor | User |
| Secondary Actors | Payment POS |
| Preconditions | User has enough money |
| Assumptions | Users will bring a unique assortment of bills. |
| Trigger | Successful Completion of Use Case: User is able to purchase products with Cash |
| Success Post Conditions | Customer purchases products with cash and receives an accurate cash difference if overpaid (i.e. paid with $30 on an order of $29 dollars, the customer should receive a dollar back). |
| Failed Post Conditions | Customer cannot purchase products with cash or receives incorrect amount of cashback |

Use Case: Card Transaction

| General Characteristics | Allow Several Forms of Currency |
| --- | --- |
| Intent | Allow Several Forms of Currency |
| Scope | Accepted Currency |
| Primary Actor | User |
| Secondary Actors | Payment POS |
| Preconditions | User has sufficient balance/funds |
| Assumptions | Must accept multiple types of cards (i.e. MasterCard, Visa, Amex, Discover) to allow as many customers to purchase items as possible |
| Trigger | Successful Completion of Use Case: Customer is able to purchase products using their card |
| Success Post Conditions | Customer purchases products with card |
| Failed Post Conditions | Customer cannot purchase products with card |

Use Case: Coupon

| General Characteristics | To offer the Customer a discount, perhaps following events and holidays, or just to clear stock. |
| --- | --- |
| Intent | Encourage shoppers to shop and spend more |
| Scope | Must be available for limited time only |
| Primary Actor | User |
| Secondary Actors | Payment POS |
| Preconditions | User has received discount |
| Assumptions | The Machine must be capable of keeping track of all currently available coupons |
| Trigger | Successful Completion of Use Case: Products price is reduced when coupon is applied |
| Success Post Conditions | Coupon is successfully applied |
| Failed Post Conditions | Coupon is unsuccessfully applied |

Use Case: Print Receipt

| General Characteristics | Print list of purchased products for customer |
| --- | --- |
| Intent | Keep track of inventory, and allow customers to keep track of their finances. |
| Scope | Print a itemized list of purchased items |
| Primary Actor | Payment POS / Inventory Database |
| Secondary Actors | User |
| Preconditions | Purchase must occur to print receipt |
| Assumptions | Not all customers will want the physical receipt and we should allow them to opt for a digital copy via email or phone number |
| Trigger | Successful Completion of Use Case: Customer is printed itemized list of products purchased and their respective prices. |
| Success Post Conditions | Receipt is successfully printed, emailed, or texted |
| Failed Post Conditions | Receipt is not successfully printed, emailed, or texted |

Use Case: Loyalty Card

| General Characteristics | Engage with Customers and allow them special privileges |
| --- | --- |
| Intent | To incentivize recurring shopping |
| Scope | Exclusive Discounts, Early Access to Sales, or a points program to game-ify the shopping experience |
| Primary Actor | User |
| Secondary Actors | Payment POS |
| Preconditions | User must be a Loyalty Member to receive loyalty card |
| Assumptions | Not every customer is a loyalty member |
| Trigger | Successful Completion of Use Case: User is given an exclusive special for their recurring shopping |
| Success Post Conditions | Loyalty card collects points at checkout |
| Failed Post Conditions | Loyalty card does not collect points. |

Use Case: Remove Item

| General Characteristics | Allow customer to edit the list of items they purchased |
| --- | --- |
| Intent | Allow for last minute adjustments |
| Scope | Must be able to remove all available items in store. |
| Primary Actor | Employee |
| Secondary Actors | Payment POS |
| Preconditions | User has requested assistance and the employee uses a card or code to access the removal tool. Must also have scanned an item first in order for the item to be removed. |
| Assumptions | Employee is on standby to aid customers |
| Trigger | Successful Completion of Use Case: Selected items are removed from list of items to be purchased |
| Success Post Conditions | Selected items are removed from list of items |
| Failed Post Conditions | Selected items are not removed from list of items |

Use Case: Request Assistance

| General Characteristics | Allow the customer to call an employee to their self checkout machine for assistance |
| --- | --- |
| Intent | Let the customers get access to help to use machine or to remove items from list of items scanned |
| Scope | Require Employee interaction |
| Primary Actor | User |
| Secondary Actors | Employee |
| Preconditions | Employee is on standby to assist |
| Assumptions | Not all customers will need assistance, and should only be used when help is needed. |
| Trigger | Successful Completion of Use Case: Employee is notified of an assistance request and the corresponding self checkout machine |
| Success Post Conditions | User is helped by employee |
| Failed Post Conditions | User is not helped by employee |

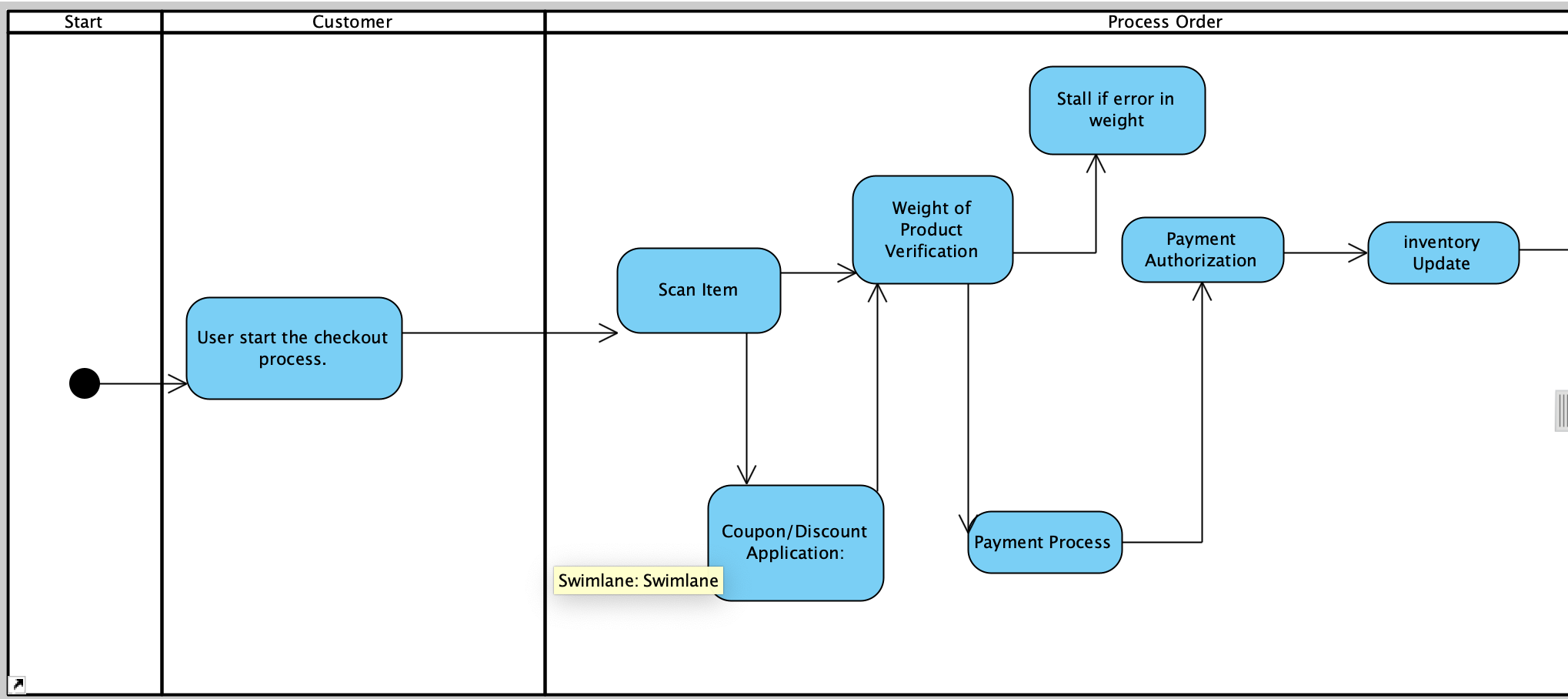
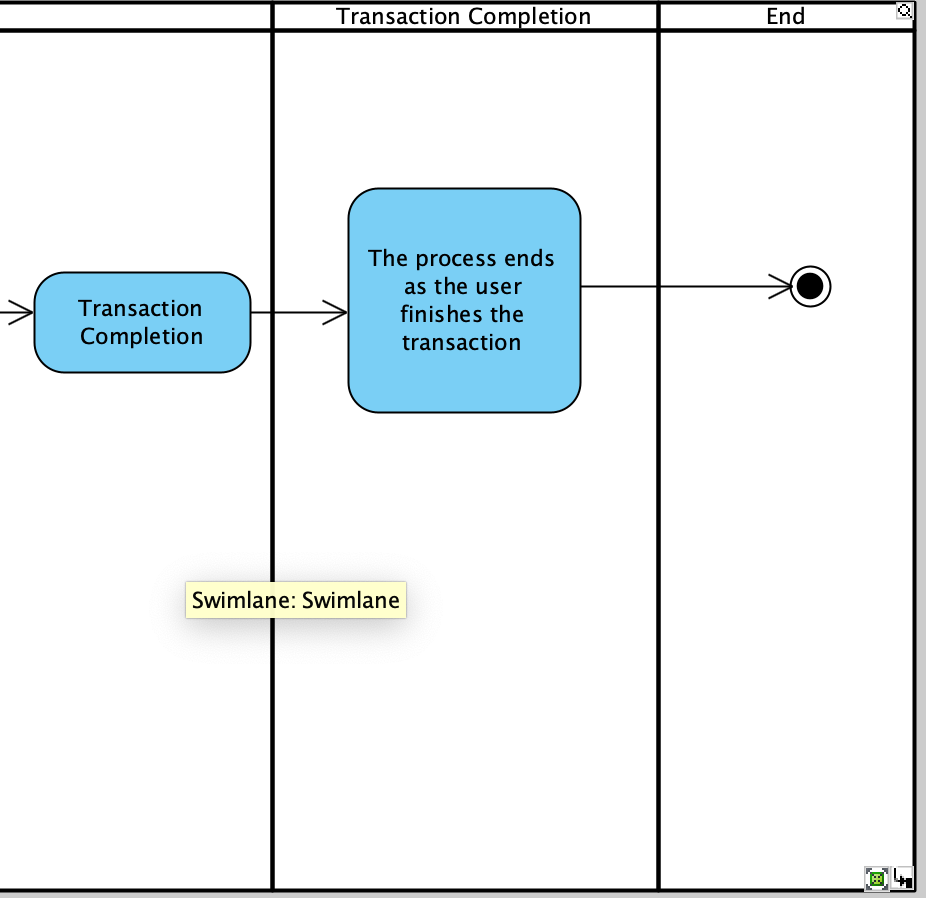
Sunny Day Scenario

| Step | Action |
| --- | --- |
| Start | The scenario begins when a customer begins scanning a transaction at the Self Checkout Machine |
| 1 | For items without barcodes, the customer is able to select the item from the inventory displayed on the screen and weigh the item on the scanner. |
| 2 | Midway through the transaction, the customer presses the “Help” button which flags down an employee for assistance in verifying the age of the customer for an alcoholic beverage. |
| 3 | The employee is able to scan their employee ID to ensure the age is verified |
| 4 | Before the employee leaves, they remove an item that was accidentally scanned twice |
| 5 | After the customer finishes scanning all the items, they scan their loyalty card to accumulate points from the purchase and apply coupons. |
| 6 | The total amount after coupons is accurate, so the customer pays with a card. |
| 7 | The payment successfully goes through, and the system prints the receipt as per the customer's request. |
| 8 | The transaction is complete. |

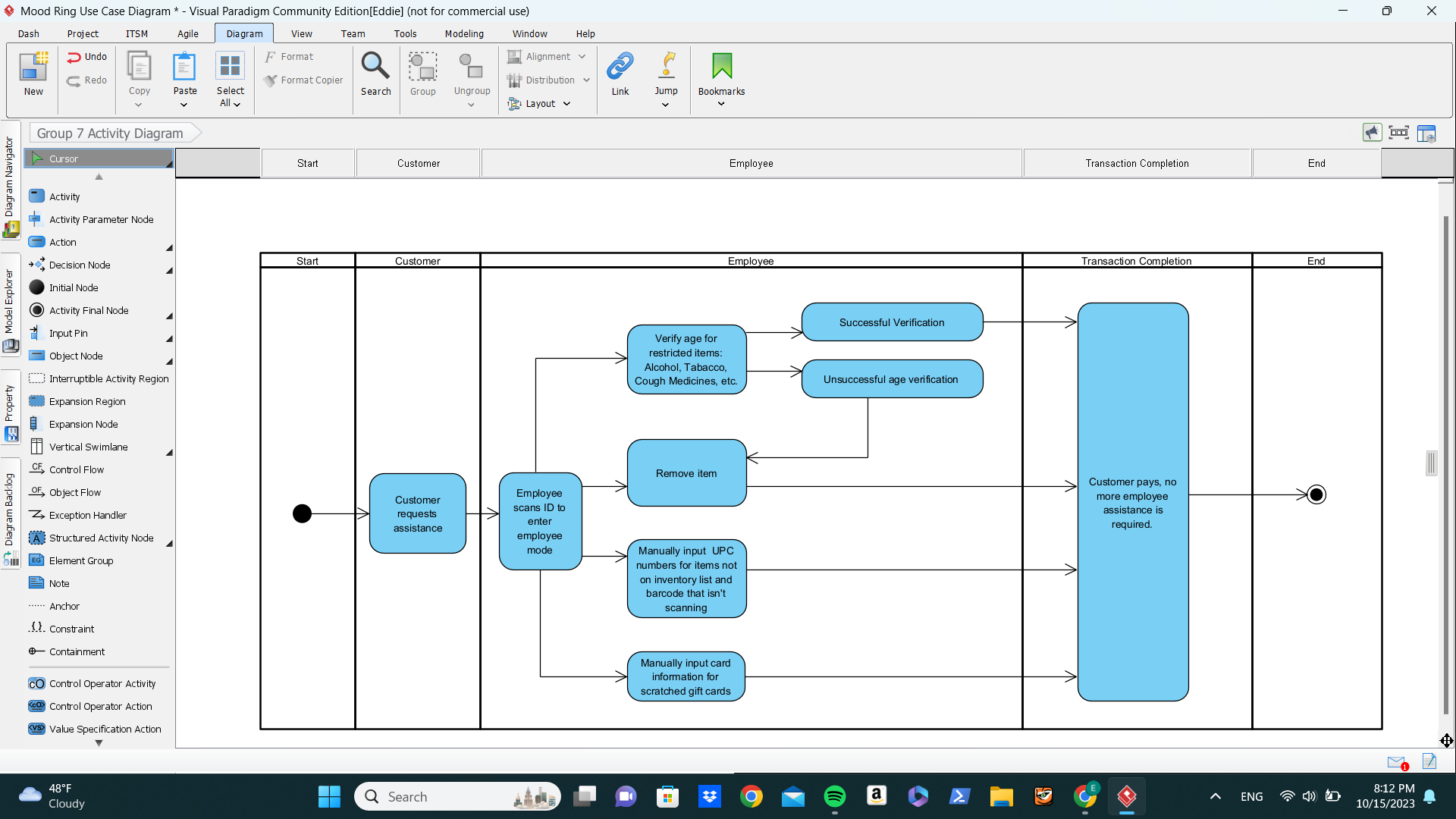
Rainy Day Scenario

| Step | Action |
| --- | --- |
| Start | This scenario begins with a failure to scan the item to begin the transaction |

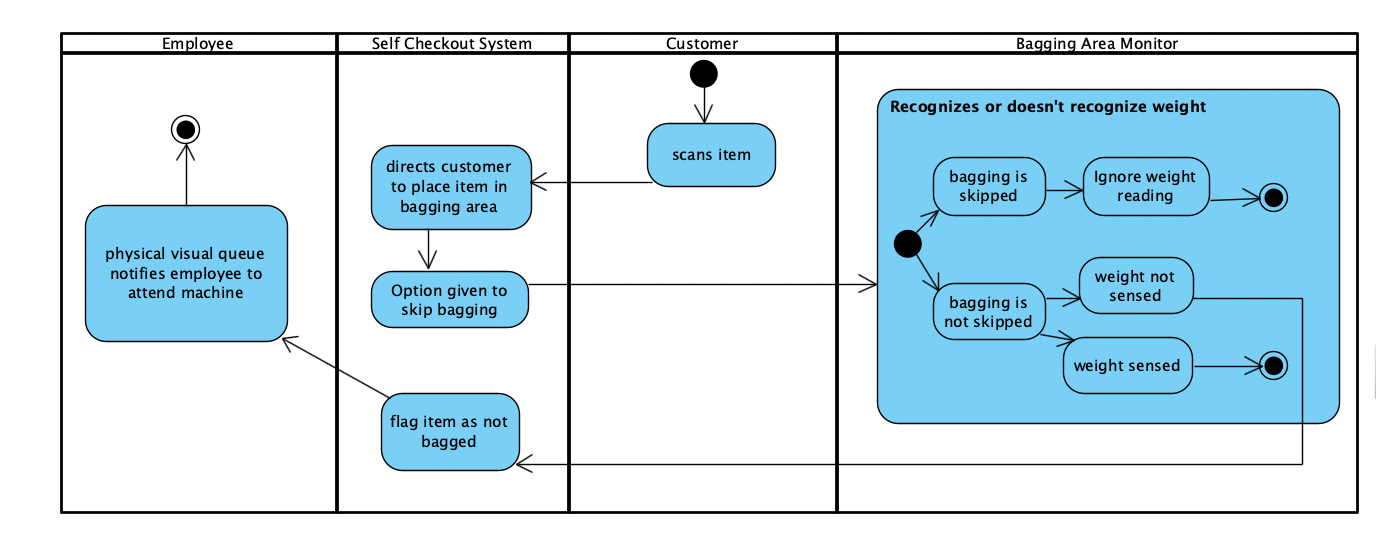
3 Specific Requirements

3.3System features

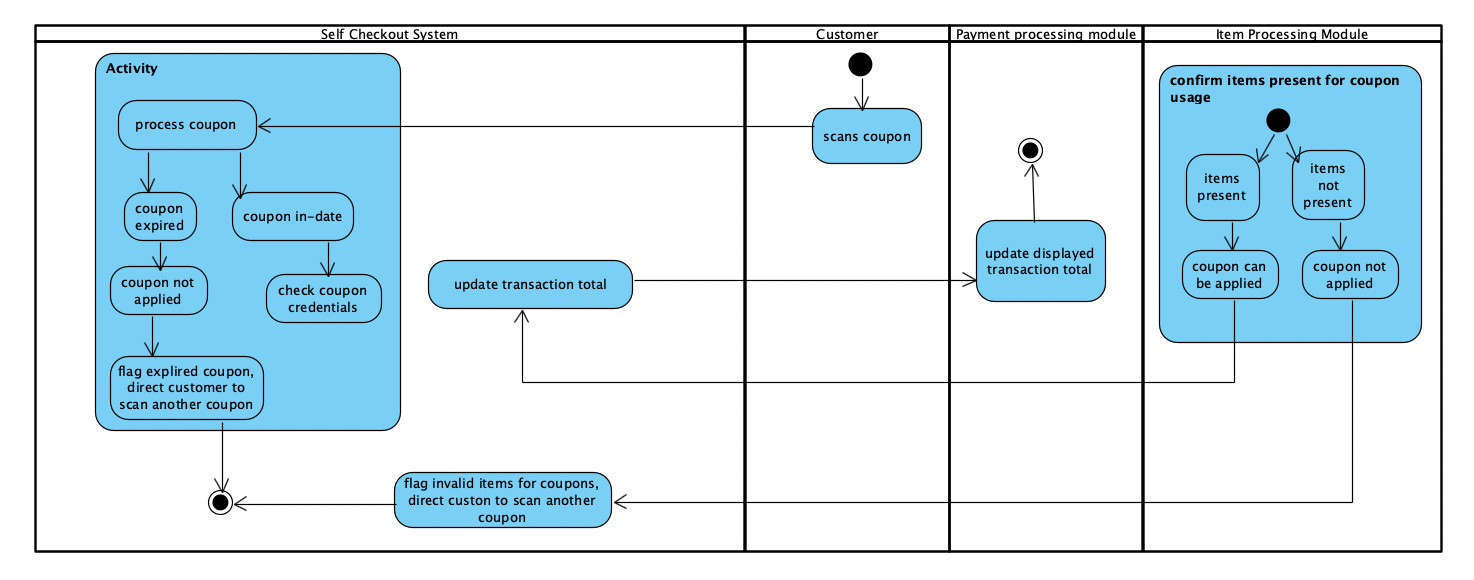
**Figure 3.1 Customer-Centered Activity Self Checkout Machine**



**Figure 3.2 Self Checkout Machine Activity**



**Figure 3.3 Self Checkout Bagging Area Activity**



**Figure 3.4 Self Checkout Coupon Activity**

**Functional Requirements:**

Checkout Process:

* The system shall allow customers to initiate the checkout process by scanning their items.
* The system shall display scanned items and their prices on the screen.

Payment Processing:

* The system shall accept credit and debit card payments.
* The system shall provide a cash payment option.
* The system shall support mobile wallet payments, such as Apple Pay and Google Pay.
* The system shall process payments securely through encryption.

Age Verification:

* The system shall prompt customers to verify their age when purchasing age-restricted items.
* The system shall accept and validate government-issued IDs for age verification.

**Non-Functional Requirements:**

Performance:

* The system shall complete payment processing within 10 seconds for 95% of transactions.
* The system shall handle a minimum of 100 concurrent users without performance degradation.

Usability:

* The user interface shall be intuitive and easy to use, with clear instructions for each step of the checkout process.
* The system shall support multiple languages for user interface text.

Reliability:

* The system shall have a maximum downtime of 1 hour per month for maintenance.
* The system shall recover from unexpected errors or crashes within 30 seconds.

Security:

* The system shall encrypt customer payment data in compliance with industry security standards.
* The system shall have user access controls to prevent unauthorized access to configuration settings.

Accessibility:

* The system shall provide accessibility features for visually impaired users, including screen reader compatibility.
* The system shall have a font size adjustment option for visually impaired users.

Scalability:

* The system shall be designed to scale up to accommodate an increased number of self-checkout machines in the future.
* The system shall have load balancing mechanisms to distribute traffic evenly.

Compliance:

* The system shall adhere to all relevant legal and regulatory requirements for payment processing and data security.

Maintainability:

* The system shall allow for software updates to be applied without affecting ongoing transactions.
* The system shall have clear documentation for maintenance and troubleshooting procedures.

Testing:

* All functional requirements shall have associated test cases to verify their proper implementation.
* Test scenarios shall be defined to validate non-functional requirements, such as performance, security, and accessibility.
* These requirements are clear, implementable, and testable, while also addressing the "shall" requirement for each statement. They cover a range of functional and non-functional aspects of the self-checkout system.