**Store Application (Point of Sale): User Document 2**

**Github:** <https://github.com/gauravshilpakar/store_java>

1. **Requirements Documentation and Specification**
   1. **Use Case Diagram**

A close up of text on a white background

Description automatically generated

* 1. **Use Case Specifications**

**Use Case Specification**

**Point of Sale System**

**Use Case Name**: Addition of Inventory

**Use Case Description:** This function allows the admin to add information about the new item received in the store. The admin can add the name, quantity, and the price of the item.

**Actors:**

* Admin

**Preconditions:**

* Items received from the vendor by admin

**Post Conditions:**

* New inventory is updated with updated quantity and price of item received
* Users has the information about availability of items in the store

**Normal Flow:**

1. The admin receives the new items in the store
2. The admin starts the application, clicks on add new inventory
3. The system checks whether the item already exists or not
4. If item exists, the system asks the number of items received
5. The user enters the number of items received
6. The system updates the inventory
7. If item doesn’t exist, the system asks the user to enter the name, quantity, and price of item.
8. The user provides the required information to the system.
9. The system updates the new inventory available with its quantity and price.

**Use Case Specification**

**Use Case Name**: Display Inventory

**Use Case Description:** This functionality allows the administrator to display the full list of items in the inventory with its price and stock quantity. On addition to this, user can also view the information about individual item as well. This method facilitates users to keep track of the item and make an order list if the item is out of stock. At the end of the day, this function is also useful for the user to provide information about what kind of service they can provide to their customer.

**Actors:**

* Admin

**Post Conditions:**

* The system displays the required items with its detail

**Normal Flow:**

1. Admin clicks on “display all item”
2. The system checks whether the database has items or not
3. If database doesn’t have the item, the system displays text “inventory empty”
4. If the database has items, the system displays all the items with its detail
5. For details of individual item, user click on that item
6. The system displays all details about that item

**Use Case Specification**

**Use Case Name**: Sell an item in inventory

**Use Case Description:** This functionality allows the user to sell an individual item from an inventory. Once the customer makes an order this function is called and the quantity of the item that is requested by the customer is deducted from the user’s inventory. If the customer requests an item that is not present in the inventory, the customer is alerted about the item being unavailable and is prompted to enter an item that the user has in the inventory. On addition to that, this function also helps the user to generate the sales report of a time (daily or monthly). User will be able to know the details about the item sold.

**Actors:**

* Admin
* Customer

**Preconditions:**

* The customer comes in the store to buy item

**Post Conditions:**

* The customer buys item from store

**Normal Flow:**

1. Customer brings an item
2. Admin scans or punch the name of item in the system
3. The system recognizes the item and displays the price of that item to the user
4. The system calculates the total amount to be charged
5. User charges the price
6. The system deducts the number of items sold from an inventory and updates inventory
7. The system generates the receipt for the item sold
8. Customer takes the receipt upon request

**Use Case Specification**

**Point of Sale System**

**Use Case Name**: Payment

**Use Case Description:** This functionality provides a payment option to the customer. It shows payment amount and allows users to choose payment options as they desire whether cash or card. It will verify the card based on the parameters provided.

**Actors:**

* Customer

**Preconditions:**

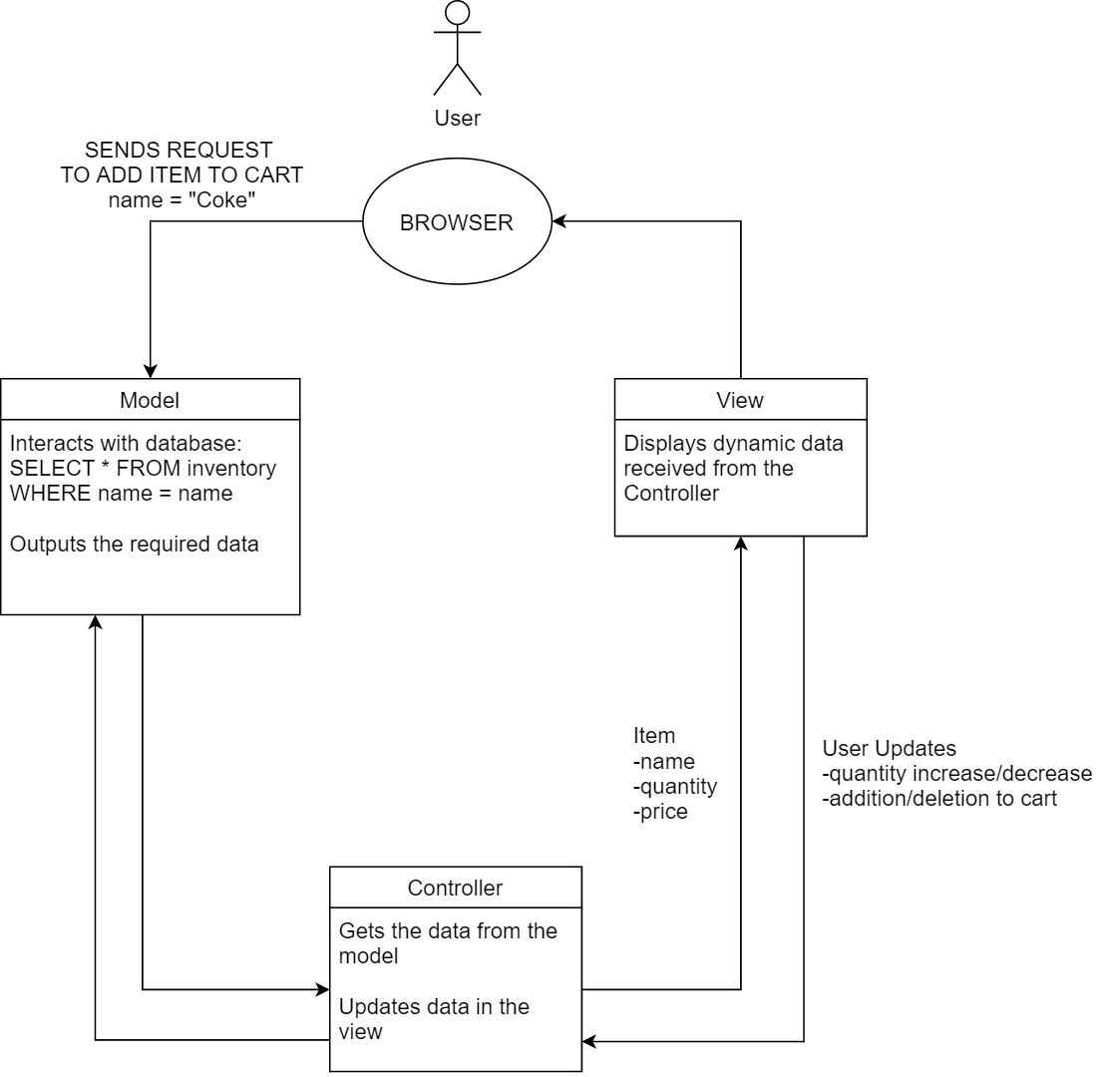
* The customer is ready to pay for the items bought from store

**Post Conditions:**

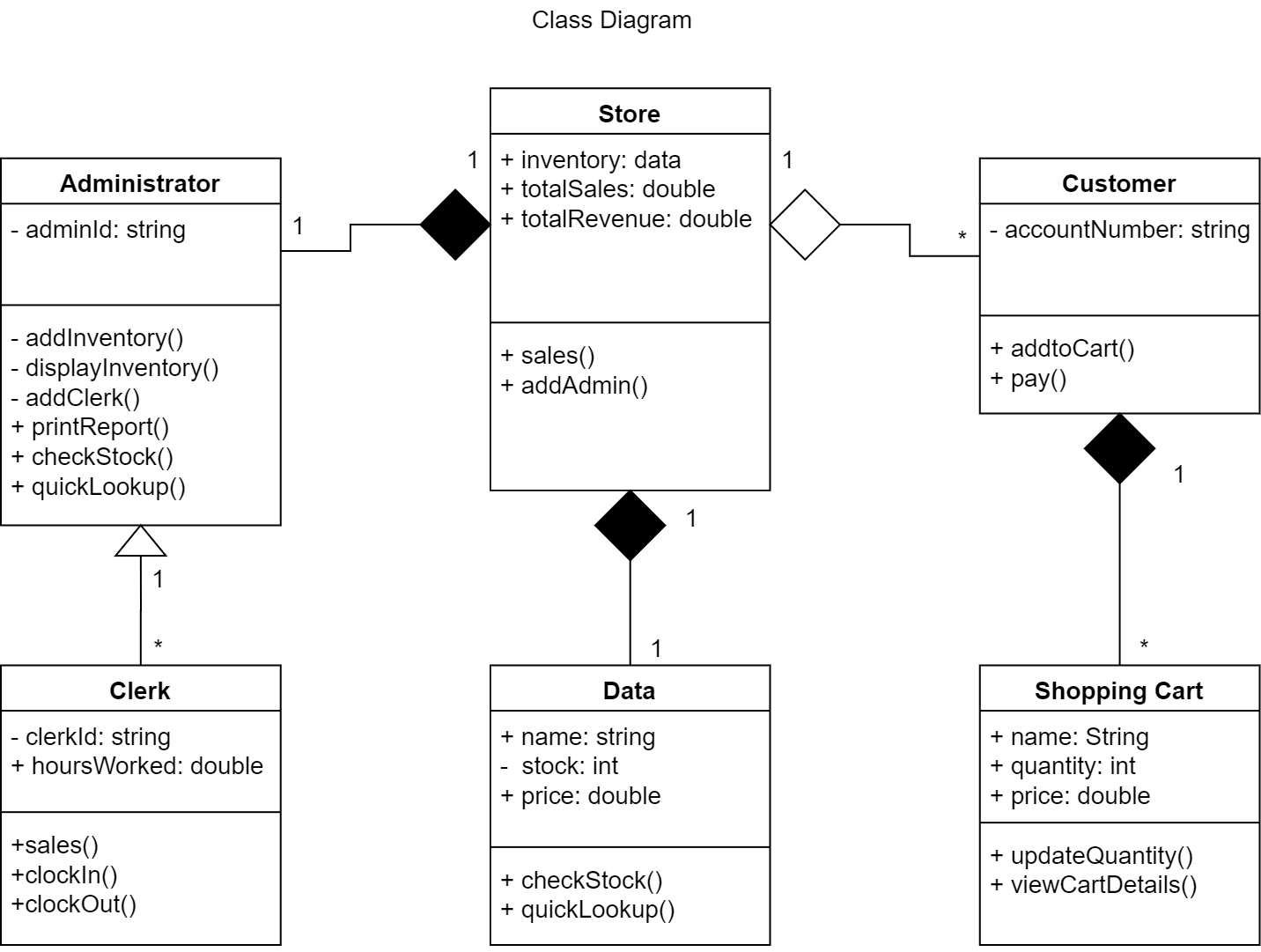
* Customer pays for the items and transaction completes

**Normal Flow:**

1. The system prompts the option CASH or CARD on customers screen
2. Customer selects the option on screen
3. If CASH is chosen, transaction is punched in by the user
4. The system generates the detail receipt including the change needed to give back
5. If CARD is chosen, the system prompts text “please insert card” to the customer
6. The customer inserts the card
7. The system verifies the card and completes the transaction
8. The system generates the detail receipt for the customer
9. **Structual Design**
   1. **Architecture Design**

****

* 1. **Class Diagram**

****

1. **Dynamic Design**
   1. **Sequence Diagrams**

A screenshot of a cell phone

Description automatically generated

Figure Adding Inventory

A screenshot of a social media post

Description automatically generated

Figure Display All Items

A screenshot of a cell phone

Description automatically generated

Figure Quicklookup

A screenshot of a social media post

Description automatically generated

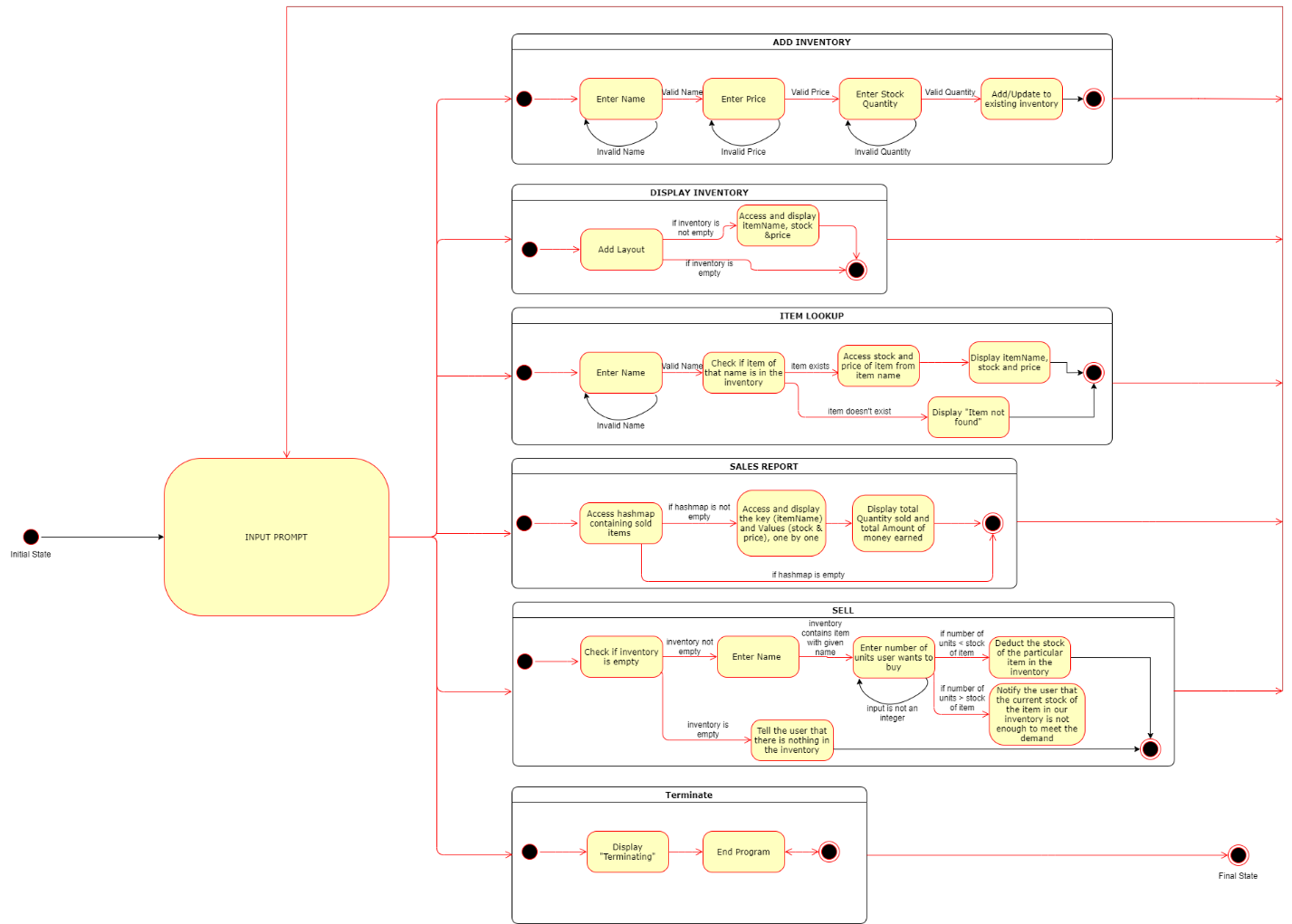
Figure Reporting

A screenshot of a cell phone

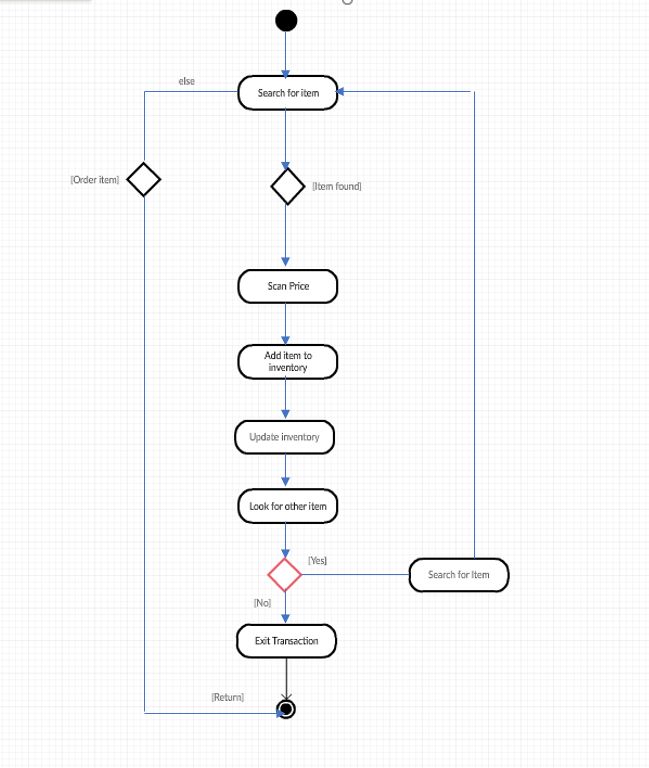
Description automatically generated

Figure Sales

* 1. **State Diagram**



* 1. **Activity Diagram (Flowchart)**



**Contributions:**

* Gaurav Shilpakar R11521961

Structural Design (Architecture Design, Class Diagram)

* Pranesh Shrestha R11521976

Sequential Diagram for each Use Case

* Raju Khadka R11536576

Requirements documentation and specification

(Use case diagram and use case specifications)

* Sudesh Subedi R11621320

Activity Diagram (Flow Chart)

* Utkrist Bhandari R11537543

Dynamic Design (State Diagram)