Supermarket Automation Software (SAS)

Software Requirements Specification(SRS)

1.Abhishek Panigrahi - 14CS10001
2.Siddhant Singh- 14CS10045

March 2016

Prepared for Software Engineering Laboratory, CS29006
Under the guidance of Dr. Partha Prathim Das

Contents

1 Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document specifies the requirements for the Supermarket Automation Software(SAS). It contains both the functional and the non-functional requirements.

1.2 Intended audience and Reading suggestions

This SRS document is intended to be read before using the software so that the user can take full advantage of this software . Dr. Partha Pratim Das ,Dr. Rajib Mall and Mr. Arpit Tarang are intended to check whether this software works correctly or not .

1.3 Scope

- The Supermarket Automation Software(SAS) is designed to support the requirements of the manager of a supermarket as given in his requirement statement .
- The manager of the supermarket wants us to develop an automation software. The supermarket stocks a set of items. Customers pick up their desired items from the different counters in required quantities. The customers present these items to the sales clerk. The sales clerk registers the items using a bar code reader and an automatic weighing scale for weighing and the data pertaining to the item and its quantity get registered in SAS.
- SAS at the end of a sales transaction prints the bill containing the serial number of the sales transaction, the name of the item, item ID, quantity, unit price, and item price. The bill should indicate the total amount payable.
- SAS should maintain the inventory of the various items of the supermarket. The manager upon query should be able to see the inventory details. In order to support inventory management, the inventory of an item should be decreased whenever an item is sold. SAS should also support an option by which an employee can update the inventory when new supply arrives.
- SAS should support printing the sales statistics for every item the supermarket deals with for any particular day or any particular period. The sales statistics should indicate the quantity of an item sold, the price realized, net sales and the profit generated.
- The manager of the supermarket should be able to change the the price at which an item is sold.

1.4 Definitions and document conventions

- Bill - A bill is a commercial document issued by a seller to the buyer indicating the products, quantities and agreed prices for products or services the seller has provided the buyer. It can indicate a sales transaction only.

- Inventory - It describes the goods and materials that a business holds for the ultimate purpose of sale.
- Bar code - A bar code is an optical machine-readable representation of data related to the object to which it is attached.
- Automatic weighing machine - An electronic device which can measure the weight of an object kept on it, and the weight is displayed on an LED display with a high level of accuracy.
- Employee - people working for the company.
- Sales clerk - A Sales clerk is an employee whose special activity includes being responsible for carrying out transactions with the customers for the different items in the supermarket.
- Manager - A Manager is the person who is responsible for supervising the employees and analyzing the sales statistics in a given period of time.
- Loose item - Items not packed e.g. fruits and vegetables.
- Packaged item - Packed items e.g. biscuit packets.
- BCR - bar code reader
- AWM/C - automatic weighing machine
- SAS - Supermarket automation software

1.5 References

IEEE The applicable IEEE standards are published in "IEEE Standards Collection," 2001 edition.

- Learning UML 2.0 by Russ Miles and Kill Hamilton.
- Fundamentals of Software Engineering by Rajib Mall

2 Overall Description

2.1 Product Perspective

The Supermarket Automation Software(SAS) has been designed to meet the growing needs of supermarkets for a faster and easy to use software to maintain the processes of billing and inventory management in the supermarket.

The Supermarket Automation Software has six active actors . Actors include sales clerk,employees,manager,automatic weighing machine(AWM/C) ,bar code reader(BCR),printer and the system server.

The use case descriptions of each actor has been described in the use case description section.

2.2 Product features

The set of functionalities that are supported by the system are documented below -

1. Perform Sales Transaction : Whenever any item is sold from the stock of the supermarket, this function prompts the clerk to pass the item over a bar code reader and an automatic weighing scale, the data regarding the item type and the quantity get automatically registered then. During the sales transaction, the name of the item, code number, quantity, unit price, and item price are entered into the bill. The bill indicates the total amount payable. The inventory is then suitably updated.

2. Read Bar code Input: Sold items are passed over the reader.
   Processing: Bar code of the item is read and the sold item is registered automatically.

3. Weigh Input: Sold items are weighed over the automatic weighing scale.
   Processing: Weight of the sold item is automatically registered.

4. Generation of the bill : A transaction bill containing the serial number of the sales transaction, the name of the items, quantity, unit price, item price and the total amount payable after adding the taxes is printed.

5. Update inventory : In order to support inventory management, this function updates the inventory whenever an item is sold. Again, when there is a new supply arrival, an employee updates the inventory level by this function.

6. Check inventory : The manager upon invoking this function issues query to view the inventory details.

7. Update prices : The manager changes the price of an item by exercising this option.

8. Print sales statistics: This option generates a printed out sales statistics for every item the supermarket deals with.

2.3 Operating Environment

The Software will operate on all Operating Systems supporting Java version 1.7 or higher.The software was originally developed on a machine running Ubuntu 14.04 LTS and Java version 1.7. On the hardware side it requires a barcode reader and a weighing machine to measure weights.

2.4 Design and Implementation Constraints

The current constraints on the project are related to the provision of database resources. The more robust and fast the database the better the performance of the software. On the hardware side the barcode reader and the weighing machine should be compatible with the operating system to operate flawlessly with the software.

2.5 Assumptions

- The manager can only invoke the system to produce the sales statistics of a given item.
- The customer can make changes during transaction but once the final bill is made the customer can't return the items back.
- The inventory of an item can be decreased only when it is sold to a customer.

2.6 Questions raised and answered

- Loose items have bar codes present on them, hence a product in the supermarket can be recognized by it's bar code.
- Sales clerk can also update product quantities on new arrival .

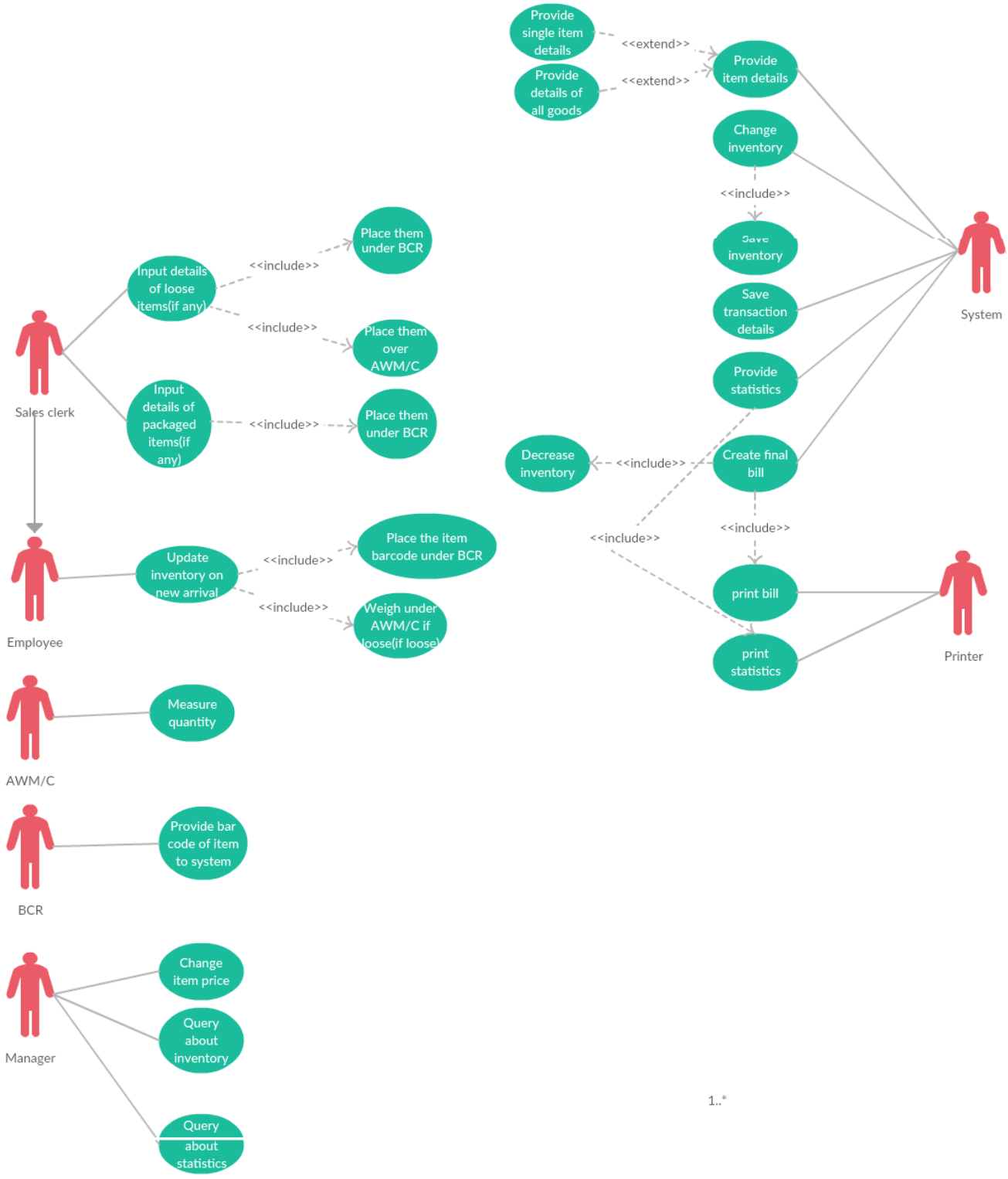

Figure 1: Use Case Diagram

3 Use Case Description

3.1 Employee

3.1.1 Use Case : update inventory on new arrival

• Brief Description

On arrival of new supply of goods in the supermarket store , the inventory needs to be updated. An employee designated to do so should be able to access the system server to make changes in the inventory.

• Inputs
New supply of goods

• Processing The employee follows the following steps.

1. The employee enters his employee id in the system server.
2. After being checked for correctness of employee id by system, the employee now can update the inventory.
3. the employee places the items received under the bar code reader one by one.
4. After this,invoking the use case "Update Inventory" with input as details of the new goods shall do the rest of the job.

• Outputs Updated inventory of goods present in the supermarket.

3.2 Sales clerk

3.2.1 Use Case : Input Details of loose items(if any)

• Brief Description

The sales clerk processes information about loose items(if present) among the goods received from a customer.

• Inputs
Selected items from user

• Processing
The sales clerk places the items under bar code reader(BCR) to capture the item bar code. Then he/she places them on the automatic weighing machine(AWM/C) to capture the quantity of the item selected by the user.

• Outputs
Quantity and item details which will be input to the "save transaction" use case of the system server.

3.2.2 Use Case : Input Details of packaged items(if any)

• Brief Description
The sales clerk processes information about packaged items(if present) among the goods received from a customer.

• Inputs
Selected items from user

• Processing
The sales clerk places the items under bar code reader(BCR) to capture the item bar code.

• Outputs
Item details which will be input to the "save transaction" use case of the system server.

3.3 Automatic Weighing Machine(AWM/C)

3.3.1 Use Case : Measure weight

• Inputs
Loose item from sales clerk

• Outputs
Weight of the loose item placed by the sales clerk.

3.4 Bar Code Reader(BCR)

3.4.1 Use Case : Measure weight

• Inputs
bar code of Item placed by sales clerk

• Outputs
Gives the item's details like product ID etc. which were stored corresponding to the given barcode.

3.5 Manager

3.5.1 Use Case : Query about inventory

• Brief Description

The manager can inquire about the goods and their prices,quantity and other properties present in the supermarket from the system server.

• Processing

The Manager first gives his/her employee id to the system server. After checking for correct employee id, the manager can ask for the complete details of all goods in the inventory by invoking the use case "Provide details of all goods " of the system.

• Outputs

Report of the details of all the goods present in the supermarket.

3.5.2 Use Case : Change prices of items

• Brief Description

The manager of the supermarket should be able to change the the price at which an item is sold as the prices of the different items vary on a day-to-day basis.

• Processing

The Manager first gives his/her employee id to the system server. After checking for correct employee id, the manager can select items whose prices need to be changed and then ask for price change of the selected items by invoking the use case "Change Inventory" of the system.

• Outputs

Changed prices of selected goods in supermarket.

3.5.3 Use Case : Query about statistics

• Brief Description

The manager can invoke the system to provide statistics of a particular item. The sales statistics includes the quantity of an item sold, the price realized, and the profit over a period of time(may be even a day).

• Processing

1. The manager first signs in using his employee id.
2. The manager can then invoke the "Provide statistics" use case of the system , giving input as item name/bar code and period of time .

• Outputs Sales statistics of the desired item.

3.6 System

3.6.1 Use Case : Save transaction details

• Brief Description
The system has to save the details of the transaction being provided by sales clerk .

• Inputs
Quantity of item(optional),bar code of item .

• Processing

1. The system first searches for item details using the use case "Search item details ".
2. The system then saves the details of the item in the transaction if the item was not already present in the transaction. Number of packets(if packaged item) should be initialized to 1.Weight of item(if loose item) should be initialized to input quantity . If the item was already present in the transaction , then in case of loose items , input weight is added to the already present weight of item otherwise just the no. of packets in the transaction is increased by 1.
3. Any changes in the transaction can be made by the sales clerk (Depending on the mood of the customer) .
4. All changes should be saved .

• Outputs
Saved transaction of goods being bought by user.

3.6.2 Use Case : Provide statistics

• Brief Description
On inquiry by manager about an item, the system should provide the statistics of the particular item. The sales statistics should indicate the quantity of an item sold, the price realized, and the profit for any particular day or any particular period .

• Inputs
Product ID or name , period of time

• Processing
The system searches for details of sale of the item over the period of time from it's database.

• Outputs
The sale statistics of the given item.

3.6.3 Use Case : Provide item data

• Brief Description
There are two cases to this use case.

1. The system itself calls this use case when it searches for the details of an item while making a transaction .
2. The manager calls this use case when he/she tries to know the details of all goods present in the supermarket.

• Inputs
Item bar code(optional :present only if call is for transaction item detail)

• Processing
There are two extension cases in this use case depending on whether call is by manager or system itself.

1. When called by manager The system simply provides it's entire database of details of goods present in the store.
2. When called by system for single item detail The system simply searches it's database using the bar code input of the item.

• Outputs
Required details of item(s).

3.6.4 Use Case : Create final bill

• Brief Description
The system now creates the final bill .It includes containing the serial number of the sales transaction,for each item : the name of the item, code number, quantity, unit price, and item price . The bill should indicate the total amount payable.
The bill should also decrease the inventory of each item as per the quantity of each item as mentioned in the transaction details.

• Inputs
Transaction details by system itself.

• Processing

1. The system creates a document where for each item : the name of the item, code number, quantity, unit price, and item price , the total amount payable is mentioned according to the transaction details as given in the input.
2. As mentioned in the transaction details , the system should decrease the inventory in it's database as per the quantity mentioned.

• Outputs The final bill to be given as input to printer to print.

3.6.5 Use Case : update inventory

- Brief Description
  There are three inner cases to this use case.

  1. The system itself calls this use case when it tries to decrease inventory after final bill has been created.

  2. The manager calls this use case when he/she wants prices of goods to change.

  3. An employee calls this use case to update inventory on new arrival.

  - Inputs

3.7 Printer

3.7.1 Use Case : print bill

- Inputs
  Bill from system server.

  - Outputs
  Printed document of the transaction made by the customer.

3.7.2 Use Case : print statistics

- Inputs
  Statistics from system server.

  - Outputs
  Printed statistics of the item under inquiry.

4 Data Flow

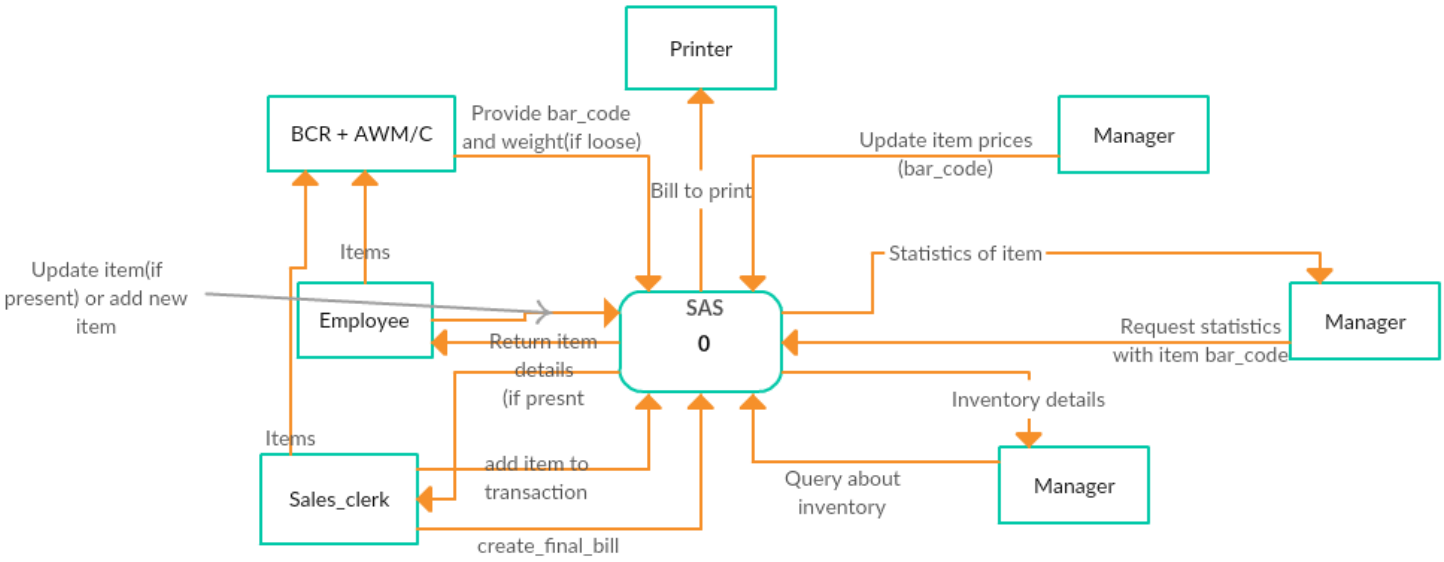

Figure 2: Context Diagram : DFD Level 0

4.1 Context Diagram

4.1.1 Brief Description

The system context diagram (SCD) defines the boundary between the system and its environment, showing the entities that interact with it.The main entities that interact with it are

1. BCR + AWM/C
2. Printer
3. Manager
4. Sales clerk
5. Employee

The various interactions between entities and the system are shown in the diagram .

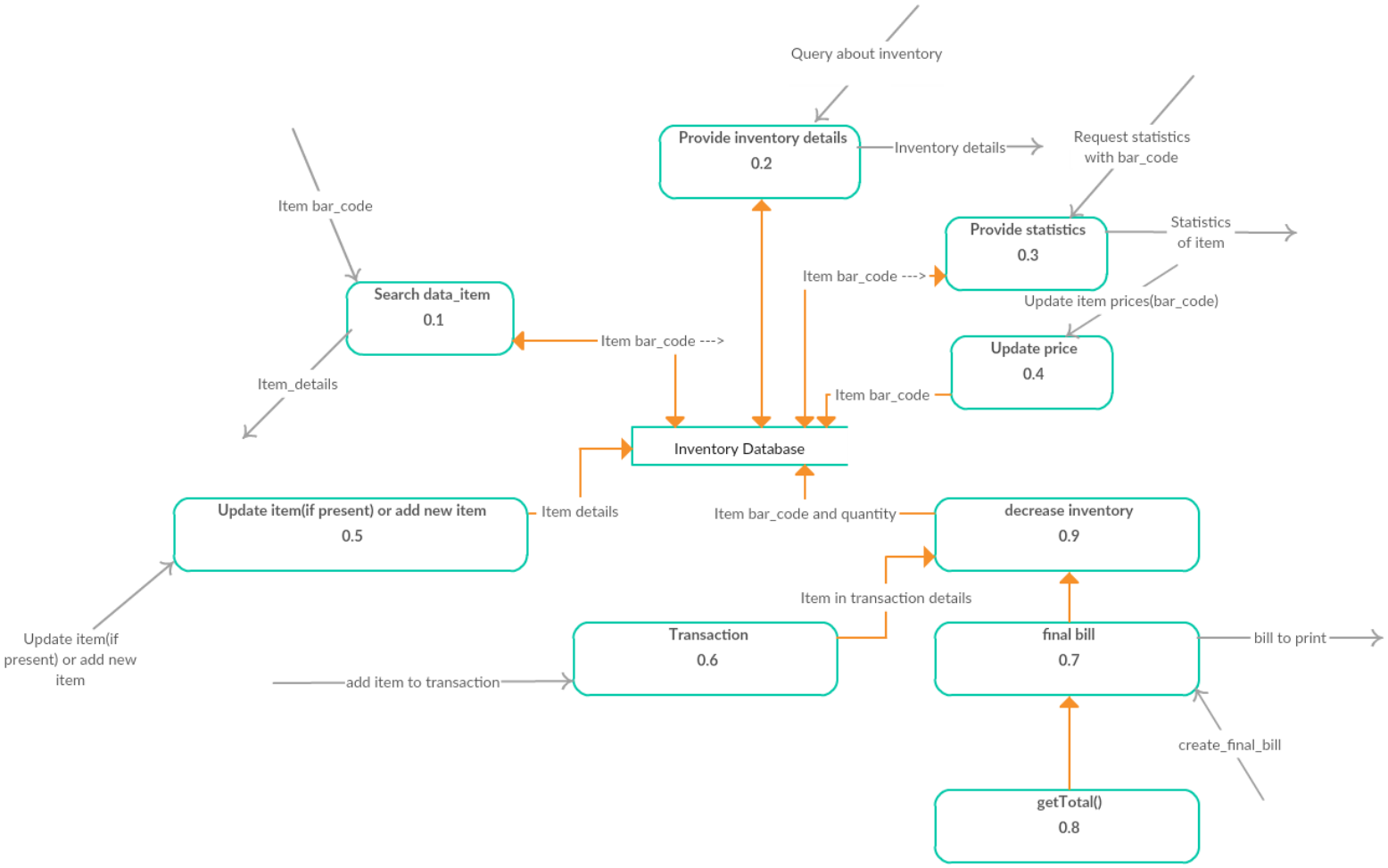

Figure 3: DFD Level 1

4.2 Data Flow Diagram Level:1

4.2.1 Brief Description

This level of DFD shows the basic processes going on inside the system that process inputs from external entities and provides output to the users . The basic processes included in the system functioning are

1. Search data item
2. Provide inventory details
3. Provide statistics
4. Update price

5. Update item(if present) or add new item
6. Transaction
7. final bill
8. getTotal
9. decrease inventory

Inventory Database supports interaction with the following processes

1. Search data item
2. Provide inventory details
3. Provide statistics
4. Update price
5. Update item(if present) or add new item
6. decrease inventory

The interactions are shown in the diagram .

5 Specific Requirements

5.1 Functional Requirements

5.1.1 Transactions

1. Inputs

   Product ID from bar code reader and weight from the weighing machine

2. Processing

   The SAS queries the database for the product information and calculates the total amount payable. A bill is then created which is printed.

3. Output

   Output is a printed bill. Contains the total amount payable apart from the details of the products and also the sales clerk.

5.1.2 Viewing sales statistics

1. Input

   Product ID and the duration for which the statistics are to be viewed.

2. Processing

   The SAS looks into the database, the cost and selling price of the particular product for every transaction during that period and generates the profit statistics.

3. Outputs

   The profit statistics are displayed in the requested format for the manager.

5.1.3 Updating prices

1. Input

   Product ID and the new price of the product.

2. Processing

   The SAS looks into the database, and displays the details of the particular product It then updates the database with the new price.

3. Outputs

   The SAS displays the details of the particular product with new price.

5.1.4 Updating inventory

1. Input

   Product ID and the quantity arrived.

2. Processing

   The SAS looks into the database, if the product already exists in the inventory database, the quantity is updated otherwise new product information has to be added to the database.

3. Outputs

   A notification is displayed confirming the update regarding the product and Quantity.

5.2 External Interface Requirements

5.2.1 User Interfaces

1. Manager interface

   The SAS screen has options to view the inventory, change the prices of the products, and view sales statistics.

2. Sales Clerk interface

   The SAS screen has options to complete a transaction with a customer and print the bill for the transaction.

3. Supermarket staff interface

   The SAS screen displays options to update the inventory with arrival of new supplies.

5.2.2 Hardware interfaces

The bar code reader scans the bar code from a packaged product and sends the product ID to the software and the weighing machine sends the weight of the product(if it is a loose product)

5.2.3 Software Interfaces

1. Inventory query
   - The manager queries the product whose details he/she wishes to view..
   - The SAS displays information about the product.
   - The manager changes the price of the product which is reflected in the database.
2. Add to inventory
   - The supermarket staff requests for the addition of the product and subsequently enters the details of the product. The product may be an existing product in the inventory whose new supply has arrived or a completely new product in itself.
   - The SAS updates the product in its database.
3. New transaction
   - The sales clerk provides the details of the product ready for billing
   - On pressing the print button, the details of the inventory are updated and a bill is produced and printed along with a confirmation message.

5.2.4 Communication interface

Any change made to the inventory of the supermarket is automatically updated in the database which has been set up in a separate server.

5.3 Non Functional Requirements

5.3.1 Performance and Reliability

High level of performance requires a good network and appropriate infrastructure for Storing large amount of data and handling multiple requests.

5.3.2 Security

The users of the software are required to login via their unique login ID and password.

5.4 Inverse Requirements

The inventory cannot be reduced without the item being purchased by a customer. Only the manager is authorised to change the price of the products

5.5 Database Requirements

All the data pertaining to employees, inventory and archive of sales information is stored in the database. For fast query this requires a robust database design. MySQL is required for maintaining the database.

5.6 Design Constraints

No significant design constraints observed.

6 Data Dictionary

- Bill = Serial number of the sales transaction + Item details(for each item) + total amount to be paid .
- Item details = Name + Code Number + Quantity + Unit Price + total Item price.
- Sales statistics of item = Quantity sold + Price realized + Profit .

.