Name: Mahim Choudhury

**Table of Contents**

[Project Direction Overview 2](#_Toc113608068)

[Use Cases and Fields 2](#_Toc113608069)

[Summary and Reflection 5](#_Toc113608070)

# Project Direction Overview

I plan to design a database to keep track of the inventories for my own business that I plan to open in the near future. I think a well versed and maintained inventory system is key run a business successfully.

A well-run inventory management system should help the manager to prevent any product shortages while not over ordering products. It ensures good customer service as the customers will always find whatever they need every time they visit the store. Also, if a store has multiple branches, a good inventory management should help the manager to order products and fill the inventory based on locations and demand. The manager can identify the most sold items or find out regular customers for future promos or sales prediction. My inventory management database should make all these jobs easier and save time for any kind of business, be it small or large.

The system will consist of an app/website that the manger or the owner can use to manage the inventory of the company/store. The app/website should only allow authorized personnel(manager/owner) to log in and perform any inventory related operation. The user of the app can store product information, track the number of items left in the inventory, record how many items are needed to fill the inventory and order them, forecast future sales, order items to fill the inventory, track the shipments, record transaction information, keep track of the returns by any customers, etc.

# Use Cases and Fields

1. The manager/owner signs up in the application and installs the application in the system

|  |  |
| --- | --- |
| *USER* | *SYSTEM* |
| *Visits the app or website* | *Displays sign up/log in page* |
| *Enters valid user information* | *Checks the information and creates the account in the database or log in the user* |

Here, the database needs to securely store some crucial information about the user who is signing up or logging in.

|  |  |  |
| --- | --- | --- |
| Field Name | What is it? | What’s it used for? |
| Employee \_ID | A unique id given to the user when they joined the company | To identify different employees or manager to avoid same name confusion |
| First\_name | User’s legal first name | Used to display the user’s name on the screen |
| Last\_name | User’s legal last name | Used to display the user’s name on the screen |
| Store\_location | The location of the store where the user will perform inventory management | This is important if the stores have multiple branches. This should avoid inventory conflicts between different branches of the business. |

1. Another usage would be to keep track of the sales and transaction made every day in a specific location

|  |  |
| --- | --- |
| *USER* | *SYSTEM* |
| *Enters the product names that were sold in one business day* | *Records the names of the products for that day* |
| *Enters the quantity of that product that was sold* | *Records the number of that specific product that was sold that day* |
| *Enters the date and transaction information* | *Keeps that date in the record as a business day and also the transaction id* |
| *Enters the location information* | *Makes sure the transaction is only for that location* |

Fields to be used in the database,

|  |  |  |
| --- | --- | --- |
| Field Name | What is it? | What’s it used for? |
| Product\_ID | A unique id for each batch of products that is coming into the warehouse. | This distinguishes different brand of products and category of products |
| Product\_Name | The name of the product | Helps to identify what product it is |
| Product\_quantity | The number of products that was sold. | To keep track of the stock |
| Sales\_date | When the product was sold. | Helps to maintain inventory and organize the restocking. |
| Transaction\_ID | A unique id generated each time the item was sold | Helps to record the unique transaction id for future returns for example. |
| Store\_location | The location of the store where the products will be arriving. | This is important if the stores have multiple branches. This should make sure the inwards are going into the correct inventory location. |

1. Next usage could be the to store the inwards of products for a particular location’s warehouse

|  |  |
| --- | --- |
| *USER* | *SYSTEM* |
| *Enters the product name and id* | *Displays the product itself* |
| *Enters the arrival date* | *Records the expected/actual date when the products should be in stock again.* |
| *Enters the location information* | *Makes sure the location information is valid and the inwards of products is only for that location’s inventory* |

Here’s the fields needed in the database,

|  |  |  |
| --- | --- | --- |
| Field Name | What is it? | What’s it used for? |
| Product\_ID | A unique id for each batch of products that is coming into the warehouse. | This distinguishes different brand of products and category of products |
| Product\_Name | The name of the product | Helps to identify what product it is |
| Product\_expected arrival date | When the product is expected to arrive in the warehouse. | User will know when the inventory would get restocked and can plan accordingly. |
| Product\_arrival date | The actual arrival date. | This will record the actual time of the arrival and help the user predict future dates of product availability and so on. |
| Store\_location | The location of the store where the products will be arriving. | This is important if the stores have multiple branches. This should make sure the inwards are going into the correct inventory location. |

1. Another key usage of this database would be to keep track of the customers information for future promotions and restocking prediction

|  |  |
| --- | --- |
| *USER* | *SYSTEM* |
| *Enters the customer’s name* | *Records the customer information* |
| *Enters the purchase date and products bought* | *Records the purchased date and the name of the product* |
| *Enters customer address if the purchase is online* | *Records the address of the customer for shipping purposes* |

Here are the fields for the database for this use case,

|  |  |  |
| --- | --- | --- |
| Field Name | What is it? | What’s it used for? |
| First\_name | Customer’s legal first name | Used to display the user’s name on the screen |
| Last\_name | Customer’s legal last name | Used to display the user’s name on the screen |
| Date of purchase | When they bought the product | Track the date of purchases to predict future purchases and maintain inventory to keep the customer returning. |
| Product\_name | What was purchased | For future stocks and promos. |
| Customer address | If the purchase was made online, this is where the products will be shipped | Shipping and promo purposes. |

1. Next usage of this database would be to keep track of the customer returns to maintain stocking of specific products.

|  |  |
| --- | --- |
| *USER* | *SYSTEM* |
| *Enters the product ID that was returned and the quantity* | *Records the product and it’s quantity that came back to stock* |
| *Enters the return reason and return date* | *Records the customer complaints and product restock date* |
| *Enters the customer information* | *Records the address, name, phone number of the customer who is returning the product* |

Here are the fields to be used in the database for this use case,

|  |  |  |
| --- | --- | --- |
| Field Name | What is it? | What’s it used for? |
| First\_name | Customer’s legal first name | Used to display the user’s name on the screen |
| Last\_name | Customer’s legal last name | Used to display the user’s name on the screen |
| In\_person\_return\_date | When they returned the product in person | Track the date of return to update the inventory stock for those products. |
| Product\_name | What was returned | For restocking |
| Customer address | Where was the product returned from | Shipment tracking and promo purposes. |
| Shipment\_tracking\_id | If the product was shipped instead of in person return, use the tracking information | To track when the return will arrive and so the user can work with the inventory accordingly. |

# Summary and Reflection

The database I am trying to design is inspired by my desire to maintain a good inventory management system for my future business. Primarily, I want this database to make any manager’s/ business owner’s life easier when it comes to managing inventory. The system if used properly, should improve inventory turnover, keep returning customers happy, help plan and predict future sales, help with cost-cutting and time saving, etc. I know a lot of the information would have to be inputted by the user i.e., daily sales information. Can the sales information be stored in another database and then this inventory database access those sales fields for its own use? Also, in the return use case mentioned in number 5, will it be useful to add return information of certain products which get returned repeatedly by either the same or different customers? Can this information be useful to identify defective products or customer fraud? What would be the best way to do it if you think it is worth adding?

Apart from those minor concerns I have noticed so far, I think this database design is very practical and useful. Although there are existing databases about inventory management, nothing beats the pleasure of developing your own inventory management system for your future business.