DATA DICTIONARY
"MARKET YARD MANAGEMENT SYSTEM"

COURSE:
IT 314 : Software Engineering

GROUP: 5.13

Abhin Kakkad	201501419
Manthan Mehta	201501420

1) Account = u_id + date + transaction_id + credit + debit + balance + transaction_type
    a) u_id = *A unique id to extract the details of the user whose account it is*
    b) date = *Date of the transaction*
    c) transaction_id = *A unique id to extract the details of the transaction*
    d) credit = *Amount credited in the account*
    e) debit = *Amount debited in the account*
    f) balance = *Balance mount in the account*
    g) transaction_type = [“Cash”| “Cheque”| “Net Banking”]
    h) update_account() : The account of the user needs to be updated when any transaction occurs

2) Goods = name + quantity + price + goods_id + owner_id
    a) name = *Name of the goods brought in*
    b) quantity = *Quantity of the goods brought in*
    c) price = *Price of the goods brought in*
    d) goods_id = *A unique id to extract the details of the goods*
    e) owner_id = *A unique id to extract the details about the seller of the goods*

3) Graphs = goods_id + price + date
    a) goods_id = *A unique id to extract the details of the goods whose graph is being plotted*
    b) price = *Price of the goods whose graph is being plotted*
    c) date = *Date for which the data is being entered for the goods*
    d) generate_graph() : To generate a graph of price fluctuations

4) Mandi Admin = admin_id + name + phone_number + email_id + password
    a) admin_id = *A unique id to extract the details of the admin*
    b) name = *Name of the Mandi admin*
    c) phone_number = *Mobile Number is used as username for login*
    d) email_id = *Email ID of the Mandi admin - used for notification purposes*
    e) password = *Password used for login*
    f) add_stock() : To add stock to the Mandi whenever any farmer brings it in
    g) update_price() : The admin enters the data about the quantity and its price on a particular day

5) Portal
    a) authenticate() : The portal would authenticate the login for any user to check if that user is present in the database and whether he has entered correct credentials
    b) take_payment() : When the goods are bought then the payment process is done by the portal
    c) notification() : Whenever a customer wants to bargain, the seller whose goods they are need to be notified of that or When a price is updated is generated then the users have to be notified of the new change
    d) verify_payment() : The portal verifies the payment made, only then will the transaction be completed
    e) checkout() : After the final deal, the user is taken to the checkout page
    f) notify_staff() : When any user buys something from the Mandi and delivery needs to be made, this function will notify the staff on duty

6) Receipt = transaction_id + entity_id + quantity + price + tax + total
    a) transaction_id = *A unique id to extract the details of the transaction*
    b) entity_id = *A unique id to extract the details of the entities involved in the transaction*
    c) quantity = *Quantity of the goods bought*
    d) price = *Price of the goods bought*
    e) tax = *Amount of tax charged on the total purchase*
    f) total = *Total amount of the purchase*
    g) calculate_tax() : After the transaction, the receipt class will calculate the tax according to the amount of goods purchased
    h) calculate_total() : After the transaction, the receipt class will calculate the total amount including the taxes and delivery charges
    i) generate_bill() : The bill needs to be printed or sent to the entities involved after the transaction is over

7) Mandi = mandi_id + admin_id + space + goods_present
    a) mandi_id = *A unique id to extract the details of the Mandi*
    b) admin_id = *A unique id to extract the details of the Mandi Admin*
    c) space = *The storage space that a Mandi has*
    d) Goods_present = *The current storage space that is occupied in the Mandi*
    e) calculate_space() : Calculate the amount of space present in the Mandi after the stock is added to it
    f) update_stock() : If any transaction is made then the stock details of the Mandi need to be updated

8) User = u_id + name + phone_number + email_id + bank_details + password + user_type
    a) u_id = *A unique id to extract the details of the user*
    b) name = *Name of the user*
    c) phone_number = *Phone Number is used as the username for login*
    d) email_id = *Email ID of the user*
    e) bank_details = *Bank Details of the user*
    f) password = *Password of the user used at the time of login*
    g) User_type = [“Customer” | “Wholesaler” | “Retailer” | “Farmer”]
    h) buy_goods() : The user would call this function when he wants to buy something
    i) sell_goods() : The user would call this function when he wants to sell something
    j) bargain() : When a user is not satisfied with the price of a commodity he/she may call this function to bargain for the price
    k) update_profile() : Any user can update his/her profile
    l) make_payment() : After the final goods are brought the user needs to make payment for the goods by any method that he wishes namely Cash, Card or Net Banking
    m) add_to_cart() : When the deal is final, the customer will add the item to his final basket

9) Mandi Staff = name + id + phone_number
    a) name = *Name of the staff person*
    b) id = *A unique id to extract the details of the staff person*
    c) phone_number = *Phone Number is used to send notifications regarding the delivery*
    d) deliver() : The staff needs to deliver the goods if the user has requested it