**Name:**  Abenes, Enrico O.  **Date:**  May 27, 2023

**Subject & Section:** CC2 – INTL 1  **Schedule:** 7: 30 am – 11: 20 am FSat

**ONLINE STORE MANAGEMENT SYSTEM**

**summary of your plan**

Creating a program that has a purpose to organize an Inventory Store and record a Sales Report using a Python/ Pycharm IDE with the assistance of the following: Algorithm, Pseudocode, and Flowchart.

When the application is run, a menu-driven interface for managing products, keeping track of inventories, handling orders, and creating sales reports is displayed. The main menu has tools for managing products (adding, updating, and removing), showing the available inventory, looking for certain products, processing customer orders, and creating sales reports. Data on product inventory and orders is loaded into the system from text files, and after any changes, the data is saved back to the original files. The system also has tools for retrieving the best-selling item and creating sales records for days, weeks, or months. Overall, the program offers a productive and systematic technique to control the inventory and sales activities of an online company.

**ONLINE STORE MANAGEMENT SYSTEM**

**algorithm**

1. **mainMenu(),** a user interface that will guide users through various options and capabilities.

A go to **manageProducts(),** if choose 1.

B go to **currentInventory(),** if choose 2.

C go to **searchProduct(),** if choose 3.

D go to **processOrder(),** if choose 4.

E go to **salesReport(),** if choose 5.

F go to **exit(),** if choose 6.

Otherwise, “invalid, try again”, if choose any number except 1-6

1. **manageProducts(),** another user interface that will guide users through various options and capabilities.

A go to **addProduct(),** if choose 1.

B go to **updteProduct(),** if choose 2.  
 C go to **removeProduct(),** if choose 3.

D go to **mainMenu(),** if choose 4.

Otherwise “invalid, try again”, if choose any number except 1-4

1. **addProduct(),** take user input for product information (name, price, and quantity) to be recorded, and save/modify the data into Inventory.txt.
2. **updateProduct(),** take user input for product information (name, new price, and new quantity) to be recorded, and save/modify the data into Inventory.txt.

If “Product Not Found”, go to **manageProducts().**

1. **removeProduct(),** take user input for product’s name, and removes it overall information from Inventory.txt.

If “Product Not Found”, go to **manageProducts().**

1. **currentInventory(),** it prints the overall list of products and its information that was recorded in Inventory.txt.
2. **searchProduct(),** take user input for product’s name, and prints its information recorded in Inventory.txt.

If “Product Not Found”, go to **mainMenu().**

1. **processOrder(),** take user input for product’s name and quantity, and it will save/modify the data in Inventory.txt and Sales.txt.

If “Not Enough Stock”/ not enough quantity, go to **mainMenu().**

If “Product Not Found”, go to **mainMenu().**

1. **salesReport(),** another user interface that will guide users through various options and capabilities and it displays the “Top Selling Product.”

A go to **dailysalesReport(),** if choose 1

B go to **weeklysalesReport(),** if choose 2

C go to **monthlysalesReport(),** if choose 3

D go to **mainMenu(),** if choose 4

1. **dailysalesReport(),** take user input for date with a format (YYYY-MM-DD), and it generates/prints the information about daily sales report within a specific day.
2. **weeklysalesReport(),** take user input for date with a format (YYYY-MM-DD), and it generates/prints the information about daily sales report within a specific week.
3. **monthlysalesReport(),** take user input for date with a format (YYYY-MM), and it generates/prints the information about daily sales report within a specific month.

**ONLINE STORE MANAGEMENT SYSTEM**

**pseudocode**

def mainMenu()

if choose ‘1’

def manageProduct()

if choose 1

def addProduct()

print(“Adding Product”)

print(“Enter Product Name: ”)

print(“Enter Price: ”)

print(“Enter Quantity: ”)

else:

print(“Product Not Found”)

if choose 2

def updateProduct()

print(“Updating Product”)

print(“Enter Product Name: ”)

print(“Enter New Price: ”)

print(“Enter New Quantity: ”)

else:

print(“Product Not Found”)

if choose 3

def removeProduct()

print(“Adding Product”)

print(“Enter Product Name: ”)

else:

print(“Product Not Found”)

elif choose 4

def mainMenu()

if choose ‘2’

def currentInventory()

print(“Current Inventory”)

print(overall list and its information)(Product Name: ‘name’, Price: ‘price’, Quantity: ‘quantity’)  
print(total price)

if choose ‘3’

def searchProduct()

print(“Tracking Inventory”)

print(“Enter Product Name: “)

print(product information (Product Name: ‘name’, Price: ‘price’, Quantity: ‘quantity’))

else:

print(“Product Not Found”)

if choose ‘4’

def processOrder()

print(“Processing Order”)

print(“Enter Product Name: “)

print(“Enter Quantity: “)

else:

print(“Not Enough Stock”)

else:

print(“Product Not Found”)

if choose ‘5’

def salesReport()

if choose ‘1’

def dailysalesReport()

print(‘daily sales report information’)

if choose ‘2’

def weeklysalesReport()

print(‘weekly sales report information’)

if choose ‘3’

def monthlysalesReport()

print(‘monthly sales report information’)

elif choose ‘4’

mainMenu()

elif choose ‘6’

exit()

**ONLINE STORE MANAGEMENT SYSTEM**

flowchart