**System Implementations**

**Recommended System Requirements**

Processors: Intel® Core™ i3 processor 4300M at 2.60 GHz.

Disk space: 4 to 8 GB.

Operating systems: Windows® 10, MACOS, and UBUNTU.

Python Versions: 3.X.X or Higher.

**Minimum System Requirements**

Processors: Intel Atom® processor or Intel® Core™ i3 processor.

Disk space: 1 GB.

Operating systems: Windows 7 or later, MACOS, and UBUNTU.

Python Versions: 2.7.X, 3.9.X.

**ACKNOWLEDGEMENT**TTT

First and foremost, praises and thanks to the God, the Almighty, for His showers of blessings throughout my research work to complete the research successfully.

We would like to express my deep and sincere gratitude to my subject teacher, **Mr. Amit Udiwal**, for giving me the opportunity to do research and providing invaluable guidance throughout this research. His dynamism, vision, sincerity and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as and honour to work and study under his guidance. We are very much thankful to our **Sr. Jasmin** for giving valuable time and moral support to develop this software. We would like to take opportunity to extend my sincere thanks and gratitude to our parents for being a source of inspiration and providing time and freedom to develop this software project. We also feel indebted to my friends for the valuable suggestions during the project work.

Ayush Malviya

[Roll No.

Class XII

**CERTIFICATE**

This is to certify that the project on ‘Inventory Management System’ is a work done by Ayush Malviya fulfilment of CBSE’S AISSCE EXAMINATION 2022-23 and has been carried out under my direct supervision and guidance. This report or a similar report on the topic has not been submitted for any other examination and does not form any other examination and does not form any other course undergone by the candidate.

Name: Ayush Malviya [Roll No.

………………….

Signature of Teacher / Guide

Name: Mr. Amit Udiwal

Designation:

………………. ….………………

**REFERENCE**

The order to work on this project on ‘Inventory Management System’ the following books & literature are referred by me during the various phrases of department of the project.

• http://www.python.org/.

• http://www.itsourcecode.org/.

• http://www.wikipedia.org/.

• Informatics Practices for Class XII

- By Sumita Arora

• Together with informatics practices.

Other than the above mentioned books, the suggestions and supervision of my teacher and my class experience also helped me to develop this software project.

**Introduction**

**Inventory Management System** is a Python code that can store and purchase thing. The sole purpose of the system is to insert, delete view, update, search  and organize the list of all things and enable we to purchase the ordered thing. The Inventory Management System is a console application Project where we can access by entering specific alphabet keys. The Inventory Management System project in python can make the management and client transactions more useful.

The Inventory Management System project having a text file to store the list of all things and automatically shows the list when the application is started. The Inventory Management System project in python also has add-to-cart functionality that helps to store temporary thing when someone order in the system.

**Objective and**

**Scope of The Project**

Inventory management is critical to a business’s success since it ensures that there is never too much or too little goods on hand, reducing the danger of stock outs and erroneous records. Inventory management allows you to save money while still meeting the demands of your customers. In other words, it provides effective operational cost control. The backbone of any business is knowing what you have, what’s in your warehouse, and how to effectively manage the supply chain.

***Features:***

* **Add an item Menu**
* **Remove an item Menu**
* **Remove specifics of an item Menu**
* **List all items Menu**
* **Inquire about an item Menu**
* **Purchase Menu**
* **Checkout Menu**

**Inventory Management System**

#Dictionaries

unit\_price={}

description={}

stock={}

#Open file with stock

details = open("stock.txt","r")

#First line of the file is the number of items

no\_items = int((details.readline()).rstrip("\n"))

#Add items to dictionaries

for i in range(0,no\_items):

line = (details.readline()).rstrip("\n")

x1,x2 = line.split("#")

x1=int(x1)

x2=float(x2)

unit\_price.update({x1: x2})

for i in range(0,no\_items):

line = (details.readline()).rstrip("\n")

x1,x2 = line.split("#")

x1=int(x1)

description.update({x1: x2})

for i in range(0,no\_items):

line = (details.readline()).rstrip("\n")

x1,x2 = line.split("#")

x1=int(x1)

x2=int(x2)

stock.update({x1: x2})

details.close()

#List to store the items purchased

cart=[]

c="y" #Runs the while loop as long as user wants

#Instructions

print("Welcome to IMS")

print()

print("A-Add an item")

print("R-Remove an item")

print("E-Edit specifics of an item")

print("L-List all items")

print("I-Inquire about a part")

print("P-Purchase")

print("C-Checkout")

print("S-Show all parts purchased")

print("Q-Quit")

print("remove-Remove an item from the cart")

print("help-See all commands again")

print()

total\_cost=0

flag=0 #To check if they have checked out

while(c!= "q" or c!= "Q"):

c= input("What would you like to do? ")

if(c=="q" or c=="Q"):

break

elif(c=="A" or c=="a"):#Add a part

p\_no = int(input("Enter part number: "))

p\_pr = float(input("Enter part price: "))

p\_desc = input("Enter part description: ")

p\_stock = int(input("Enter part stock: "))

m=0

for i in range(0,len(unit\_price)):

if(p\_no in unit\_price):

p\_no+=1

m=1

if(m==1):

print()

print("That part number already exists :(, changing value to ",p\_no)

unit\_price.update({p\_no: p\_pr})

description.update({p\_no: p\_desc})

if(p\_stock > -1):

stock.update({p\_no: p\_stock})

else:

p\_stock = 0

stock.update({p\_no: p\_stock})

print("The stock of an item cannot be negative, the stock has been set to 0.")

print()

print("Part number: ",p\_no," Description: ",description.get(p\_no)," Price: ",unit\_price.get(p\_no)," Stock: ",stock.get(p\_no))

print("Part was added successfully!")

print()

elif(c=="E" or c=="e"):#Edit a part

print()

p\_no = int(input("Enter part number: "))

if(p\_no in unit\_price):

p\_pr = float(input("Enter part price: "))

p\_desc = input("Enter part description: ")

p\_stock = int(input("Enter part stock: "))

unit\_price.update({p\_no: p\_pr})

description.update({p\_no: p\_desc})

stock.update({p\_no: p\_stock})

else:

print("That item does not exist, to add an item use a")

print()

elif(c=="R" or c=="r"):#Remove a part

print()

p\_no = int(input("Enter part number: "))

if(p\_no in unit\_price):

are\_you\_sure = input("Are you sure you want to remove that item(y/n)? ")

if(are\_you\_sure=="y" or are\_you\_sure=="Y"):

unit\_price.pop(p\_no)

description.pop(p\_no)

stock.pop(p\_no)

print("Item successfully removed!")

print()

else:

print("Sorry, we don't have such an item!")

print()

elif(c=="L" or c=="l"):#List all the parts

print()

print("Parts and their prices: ",unit\_price)

print("Descriptions: ",description)

print("Stock left of Item: ",stock)

print()

elif(c=="I" or c=="i"):#Inquire about a part

print()

p\_no=int(input("Enter Part Number: "))

if(p\_no in unit\_price):

print()

print("Part number: ",p\_no," Description: ",description.get(p\_no)," Price: ",unit\_price.get(p\_no)," Stock: ",stock.get(p\_no))

if(stock.get(p\_no)<3 and stock.get(p\_no)!=0):

print("Only ",stock.get(p\_no)," remaining! Hurry!")

print()

else:

print("Sorry we don't have such an item!")

print()

elif(c=="P" or c=="p"):#Purchase a part

print()

p\_no = int(input("Enter Part number: "))

if(p\_no in unit\_price):

if(flag==1):

flag=0

stock\_current = stock.get(p\_no)

if(stock\_current>0):

stock\_current = stock.get(p\_no)

stock[p\_no] = stock\_current-1

item\_price = unit\_price.get(p\_no)

total\_cost = total\_cost+item\_price

print(description.get(p\_no),"added to cart: ","$",item\_price)

cart.append(p\_no)#Stores item in cart

else:

print("Sorry! We don't have that item in stock!")

else:

print("Sorry! We don't have such an item!")

print()

elif(c=="C" or c=="c"):#Check out

print()

print("You bought the following parts: ",cart)

print("Total: ","$",round(total\_cost,2))

tax= round(0.13\*total\_cost,2)

print("Tax is 13%: ","$",tax)

total = round(total\_cost+tax,2)

print("After Tax: ","$",total)

total\_cost=0

flag=1

print()

print("You can still purchase items after check out, your cart has been reset. To quit press q")

print()

elif(c=="help"):#Display all commands

print()

print("Help Centre")

print("A-Add an item")

print("R-Remove an item")

print("E-Edit specifics of an item")

print("L-List all items")

print("I-Inquire about a part")

print("P-Purchase")

print("C-Checkout")

print("S-Show all parts purchased")

print("remove-Remove an item from the cart")

print("help-See all commands again")

print("If you have any other questions or concerns please contact the manager.")

print()

elif(c=="remove" or c=="Remove"):#To remove an item from the cart

print()

are\_you\_sure = input("Are you sure you want to remove an item from the cart(y/n)? ")

if(are\_you\_sure=="y"):

p\_no = int(input("Enter part number to remove from cart: "))

if(p\_no in cart):

stock\_current = stock.get(p\_no)

stock[p\_no] = stock\_current+1

item\_price = unit\_price.get(p\_no)

total\_cost = total\_cost-item\_price

j=0

for i in range(0,len(cart)):#To find the index of the part in the list cart

if(i==p\_no):

j=i

cart.pop(j)

print(description.get(p\_no),"removed from cart: ")

print()

else:

print()

print("That item is not in your cart!")

print()

elif(c=="s" or c=="S"):#prints list cart

print()

print(cart)

print()

else:

print()

print("ERROR! Contact manager for help!")

print()

#Outputs total if the user quits without checking out

if(total\_cost>0 and flag==0):

print()

print("You bought: ",cart)

print("Total: ","$",round(total\_cost,2))

tax= round(0.13\*total\_cost,2)

print("Tax is 13%: ","$",tax)

total = round(total\_cost+tax,2)

print("After Tax: ","$",total)

print()

print("Thank you for using IMS")

#Write the updated inventory to the file

details = open("stock.txt","w")

no\_items=len(unit\_price)

details.write(str(no\_items)+"\n")

for i in range(0,no\_items):

details.write(str(i+1)+"#"+str(unit\_price[i+1])+"\n")

for i in range(0,no\_items):

details.write(str(i+1)+"#"+description[i+1]+"\n")

for i in range(0,no\_items):

details.write(str(i+1)+"#"+str(stock[i+1])+"\n")

details.close()