PROJECT

Factory Management System

-Vedant Gupta(R.no.28)
Aditya Singh(R.no.3)

INDEX

Certificate

COMPUTER PROJECT

This is to certify that

Vedant Gupta and Aditya Singh

Have successfully completed the computer-science project of

FACTORY-MANAGAMENT SYSTEM

SUJATA BHARDWAJ

TEACHER

ACKNOWLEDGEMENT

This project (Factory Database Management System) was made by Aditya Singh and Vedant Gupta. This project is copyrighted and plagiarizing this would be a punishable offence.

Aditya Singh

Vedant Gupta

AIM

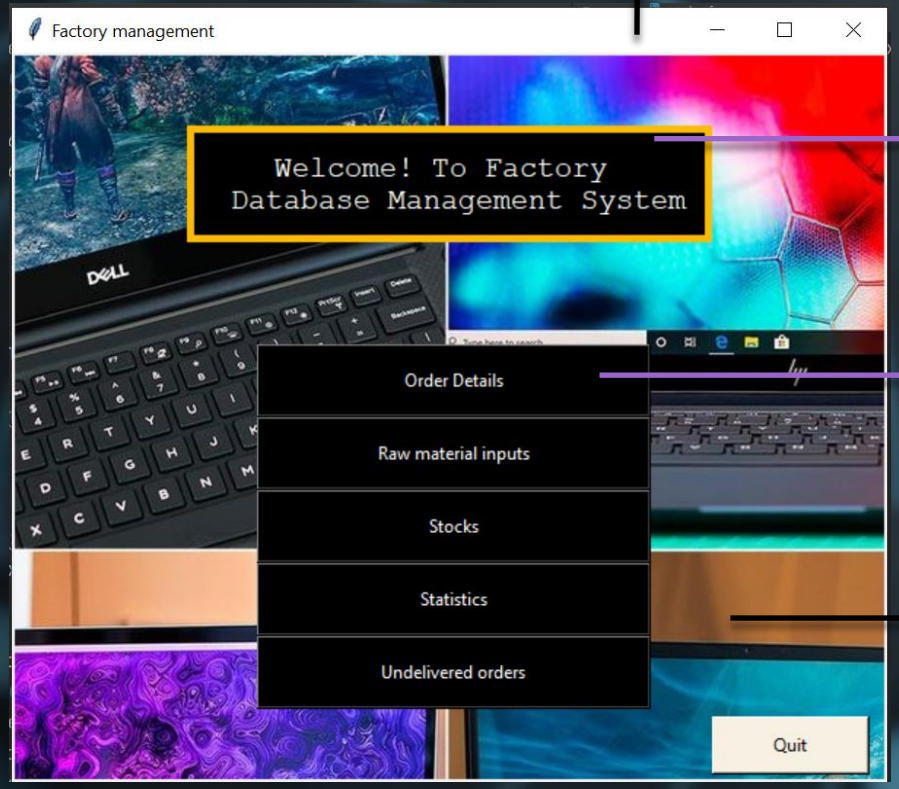
- A factory management system to manage the input, output and to have an overview of the work going on inside the premises. It tells us the order details along with the client details, the raw inputs, stocks, business statistics etc. thus making it easy for a person to manage the overall word.

ANALYSIS

- The system is based on manipulation of lists.
- Python modules such as Tkinter, pillow, CSV have been used so as to make the application user friendly.
- With the help of MySQL, the data is stored on server from which it can be accessed anywhere anytime with the application.

DESIGNING

- **Tkinter**- Tkinter is a library that provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit. It has many widgets such as buttons, canvas, checkboxes etc.



GUI Window

Heading label

Buttons

Canvas

- **GUI Window:** Creating a GUI application using Tkinter is an easy task. All you need to do is perform the following steps –

- Import the Tkinter module.
- Create the GUI application main window.
- Now you can add widgets of your choice.
- For e.g.

```
root = Tk()
root.title("Factory management")
root.minsize(width=400, height=400)
root.geometry("600x500")
background_image = Image.open("laptops.jpg")
```

- **Heading label:** This widget implements a display box where you can place text or images. The text displayed by this widget can be updated at any time you want.

- For e.g.

```
headingFrame1 = Frame(root, bg="#FFBB00", bd=5)
headingFrame1.place(relx=0.2, rely=0.1, relwidth=0.6, relheight=0.16)
headingLabel = Label(headingFrame1, text="Welcome! To Factory \n Database Management System", bg='black', fg='white',
                     font=('Courier', 15))
```

- **Buttons:** The Button widget is used to add buttons in a Python application. These buttons can display text or images that convey the purpose of the buttons. You can attach a function or a method to a button which is called automatically when you click the button.

- For e.g.

```
btn2 = Button(root, text="Raw material inputs", bg='black', fg='white', command=rawmaterials)
btn2.place(relx=0.28, rely=0.5, relwidth=0.45, relheight=0.1)
```

- **Canvas:** The Canvas is a rectangular area intended for drawing pictures or other complex layouts. You can place graphics, text, widgets or frames on a Canvas.

- PARAMETERS:

a) Bd - Border width in pixels. Default is 2.

b) Bg - Background colour.

c) Height - Size of the canvas in Y dimension.

d) Width - Size of the canvas in X dimension.

```
Canvas1 = Canvas(root)
Canvas1.create_image(300, 340, image=img)
Canvas1.config(bg="white", width=newImageSizeWidth, height=newImageSizeHeight)
Canvas1.pack(expand=True, fill=BOTH)
```

➢ **Pillow**- Pillow is one of the important modules for image processing in python.

- Image Archive: The Python Imaging Library(PIL) is best suited for image archival and batch processing applications. Python pillow package can be used for creating thumbnails, converting from one format to another and print images, etc.
- Image Display: You can display images using Tk PhotoImage, BitmapImage and Windows DIB interface, which can be used with PythonWin and other Windows-based toolkits and many other Graphical User Interface (GUI) toolkits.
- Image Processing: The Pillow library contains all the basic image processing functionality. You can do image resizing, rotation and transformation.

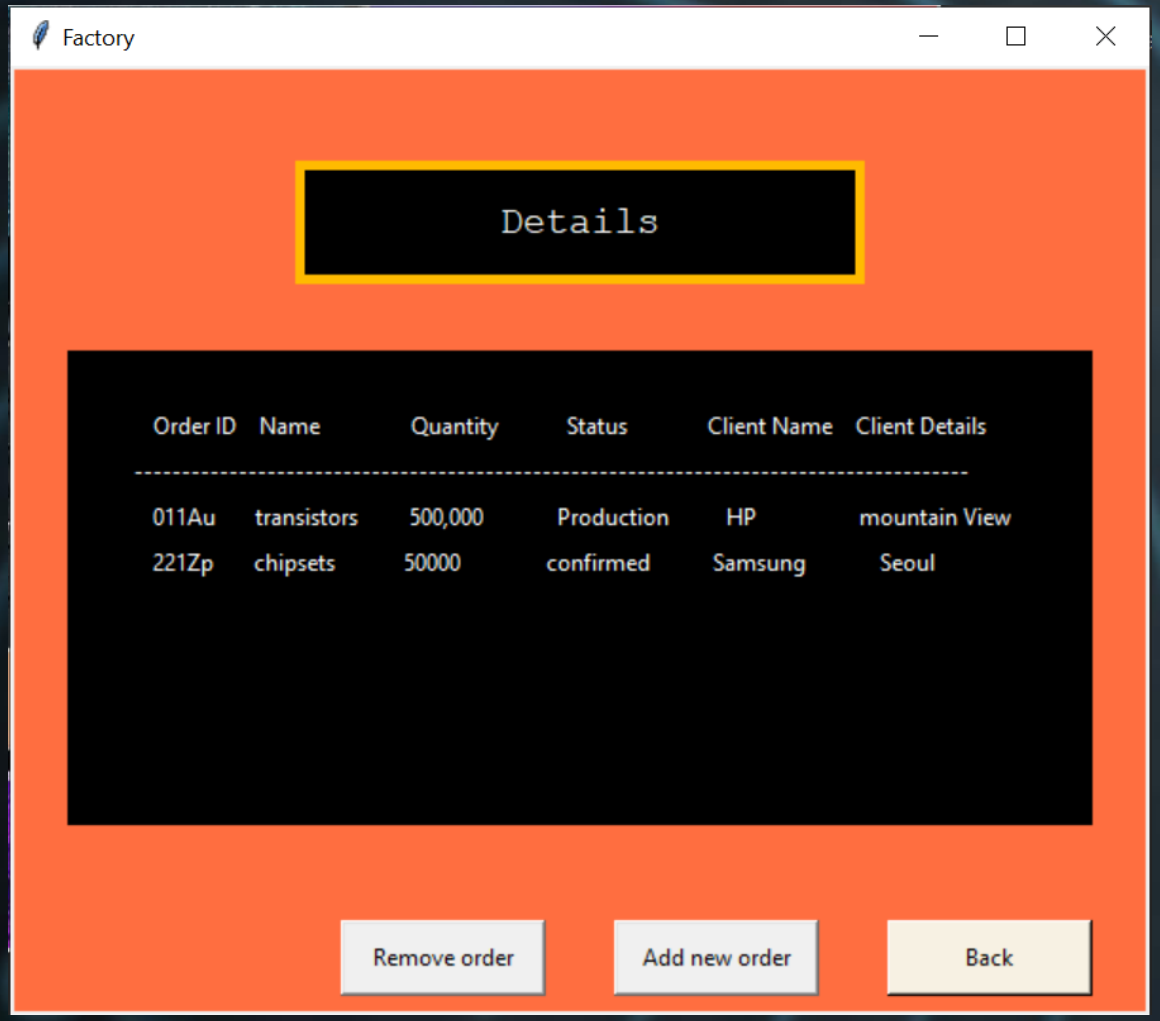
```
background_image = Image.open("laptops.jpg")
[imageSizeWidth, imageSizeHeight] = background_image.size

newImageSizeWidth = int(imageSizeWidth * n)
if same:
    newImageSizeHeight = int(imageSizeHeight * n)
else:
    newImageSizeHeight = int(imageSizeHeight / n)

background_image = background_image.resize((newImageSizeWidth, newImageSizeHeight), Image.ANTIALIAS)
img = ImageTk.PhotoImage(background_image)
```

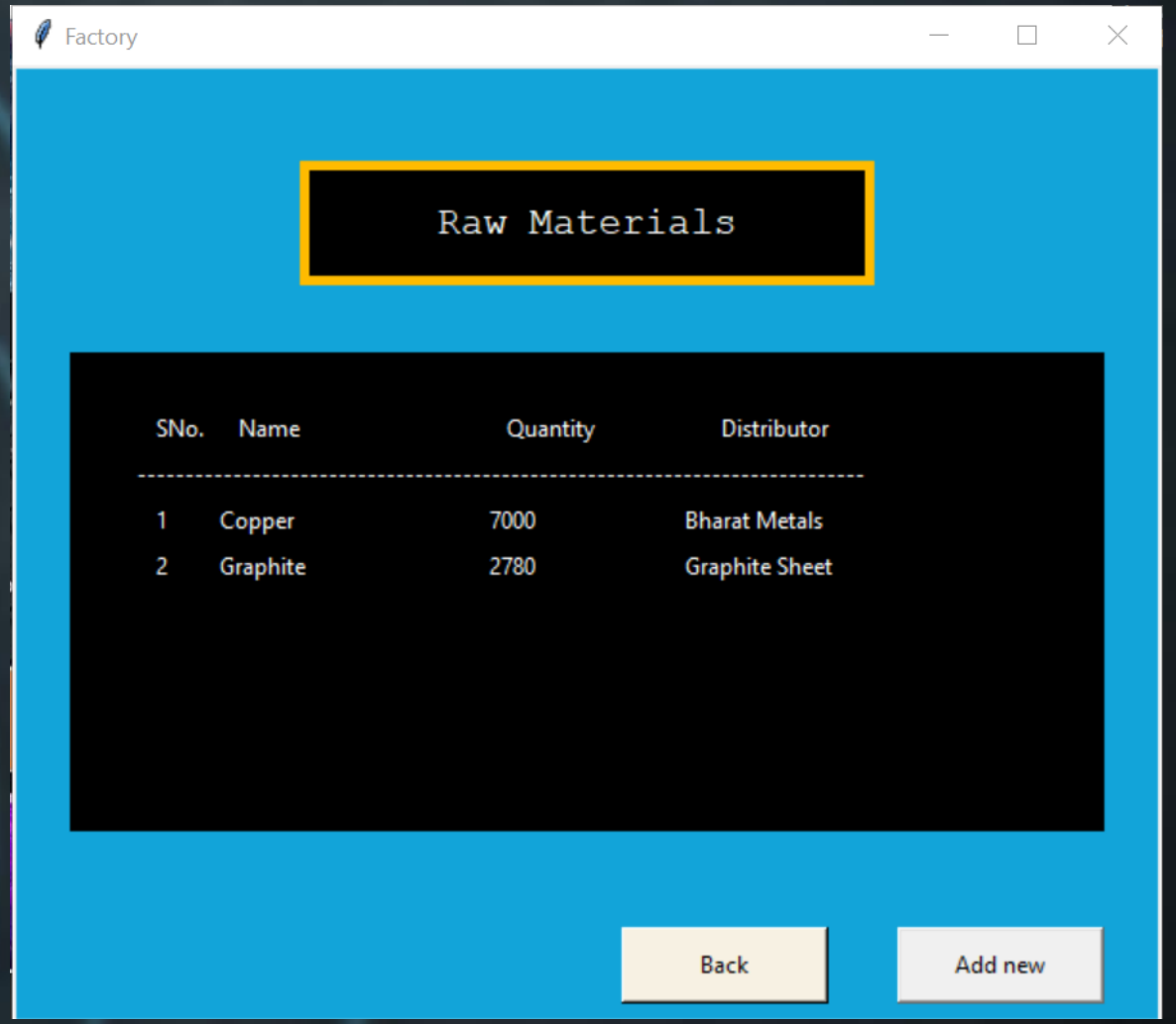
- **CSV and Message box:** The CSV file or comma separated values file are one of the most widely used flat files to store and hare data across platforms.

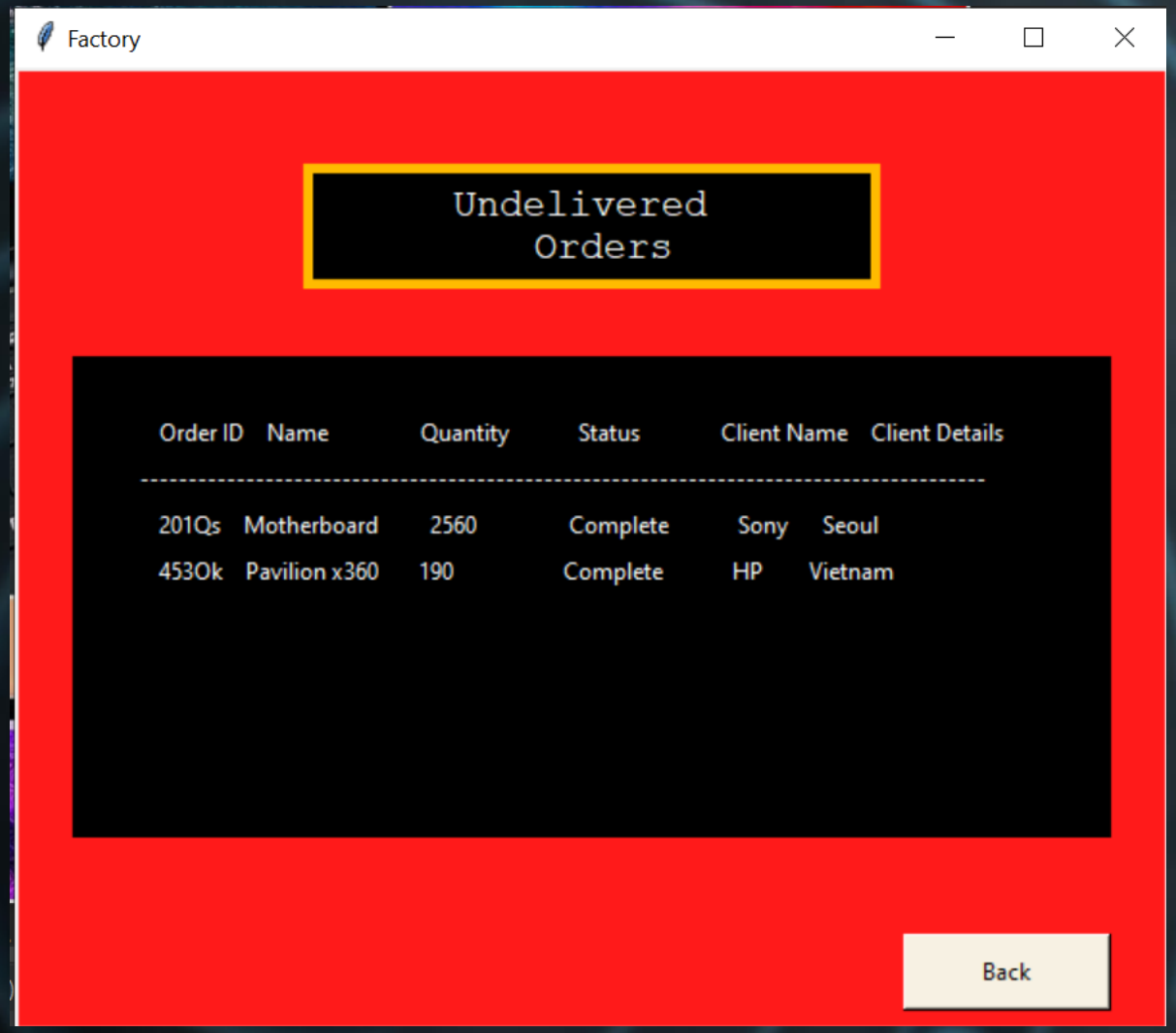
```
with open('test.csv') as f:
    reader = csv.DictReader(f, delimiter=',')
```

```
def success():
    messagebox.showinfo('Task initiated successfully')
```

OUTPUT


Factory
Details
Order ID Name Quantity Status Client Name Client Details
011Au transistors 500,000 Production HP mountain View
221Zp chipsets 50000 confirmed Samsung Seoul
Remove order
Add new order
Back


Factory
Raw Materials
SNo. Name Quantity Distributor
1 Copper 7000 Bharat Metals
2 Graphite 2780 Graphite Sheet
Back
Add new


Factory
Undelivered Orders
Order ID Name Quantity Status Client Name Client Details
201Qs Motherboard 2560 Complete Sony Seoul
453Ok Pavilion x360 190 Complete HP Vietnam
Back


Factory
Statistics
Check the latest Company statistics here!!
Back
Governance Factors | Positive Econ. Profit | PN 4 Companies
Debt Amt. | R 184.92(496.52) | R 620.96(1684.53)
Equity Mkt. Value | R 873.24(2248.79) | R 83.17(94.85)
Sales Growth | 3.14%(0.34) | 17.80%(1.36)
Asset | 8.25%(0.44) | 0.01%(0.45)
Sales | R 787(1339) | R 3392(13834)
Valuation Factors | R 554(609) | R 919(1520)
Disclosure Index | 41%(7) | 46%(9.1)
Comp. Comm. | 16 | 7
Big 4 Audit | 13 | 13

SCOPE OF IMPROVEMENT

- SQL integration
- Addition and storage of new data.

BIBLIOGRAPHY

- www.tutorialspoint.com
- data-flair.training