School: SOET Campus: VZM

Academic Year: 2024/2025 Subject Name: DAVP Subject Code: CVTM1018

Semester: 7th Program: B.TECH Branch: ECE Specialization: ECE

Date:

Applied and Action Learning

(Learning by Doing and Discovery)

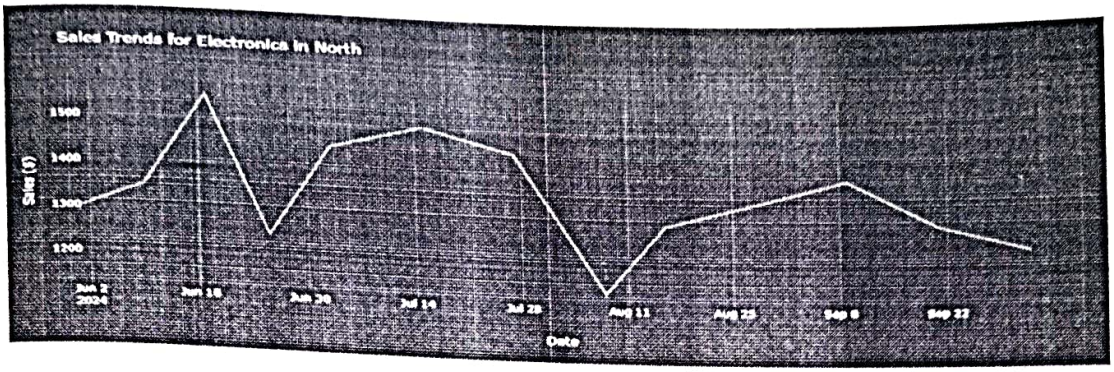
me of the Experiement : Design an interactive dashboard to visualize sales trends, allowing users to filter by region & product category

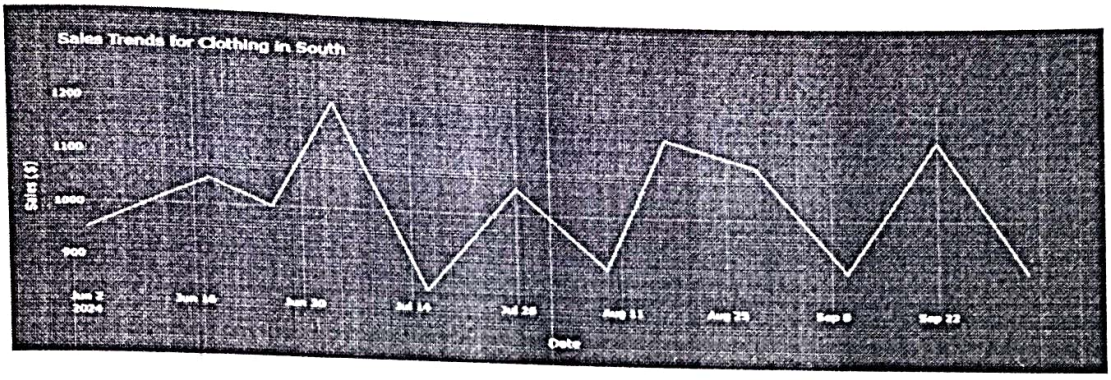
oding Phase: Pseudo Code / Flow Chart / Algorithm

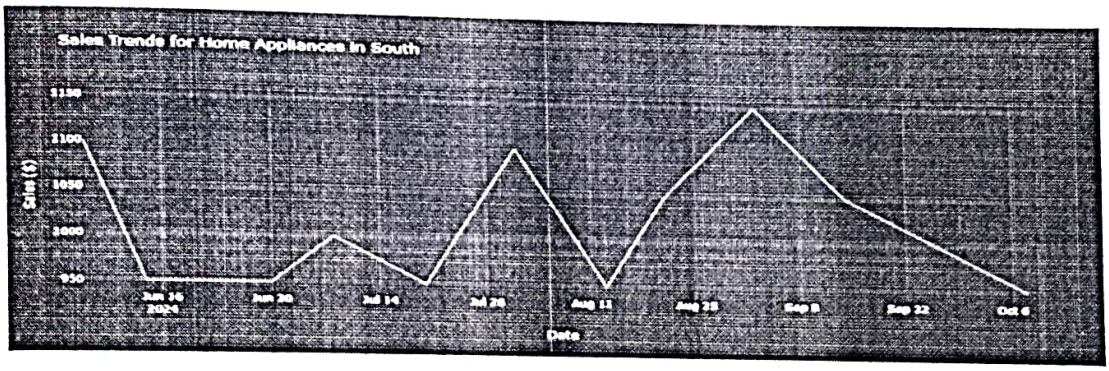
- import required libraries for building the interactive dashboard.
- load the sales data csv and inspect the structure to ensure the correct columns and data format.
- Rename columns if necessary & convert the 'Date' column to date-time format for accurate plotting
- Define the app layout with dropdown filters for selecting the region and product category, along with a line chart
- Set up a callback function to filter the data based on user selections & update the line.
- Run the app in Jupyter notebook mode using 'app.run-server' to display the dashboard.

esting Phase: Compilation of Code (error detection)

Implementation Phase: Final Output (no error)







ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10		
Planning and Execution/ Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	10		
Total	**50**		

Signature of the Student: G. Sumanth

Name : G. Sumanth

Regn. No. : 211801131001

Signature of the Faculty:

```
Import dash
from dash import dcc, html
import plotly.graph_objects as go
import pandas as pd
df = pd.read_csv("product_data.csv")
required_columns = ['Product', 'Sales', 'Profit']
missing_columns = [col for col in required_columns if col not in df.
if missing_columns:
    raise ValueError(f"The dataset is missing the following
                     required columns: {missing_columns}")
app=dash.Dash(__name__)
Pie_chart = go.figure(data = [go.pie(
    labels = df['Product'],
    values = df['Sales'],
    title = "Market Share by Product",
    hole = 0.3
)])
bar_chart = go.figure(data=[go.Bar(
        x = df['Product'],
        y = df['Sales'],
    )])
```

```
bar_chart.update_layout(
    title="Sales by Product",
    xaxis_title="Product",
    yaxis_title="Sales"
)
scatter_plot = go.Figure(data=[go.Scatter(
x=df['Product'],
y=df['profit'],
mode='markers',
markers=dict(size=12)
)])
scatter_plot.update_layout(
        title="Profits by product",
        xaxis_title="product",
        yaxis_title="profit",
    )
app.layout = html.Div([
    html.H1("Sales Dashboard", style={'textAlign': 'center'}),
    html.Div([
        html.Div([
            dcc.Graph(figure=pie_chart),
        ], classname='six columns'),
        html.Div([
            dcc.Graph(figure=bar_chart),
        ], classname='six columns'),
    ])
app.run_server(debug=True, port=8051)
```