Tailwind Traders Sales and Profit Analysis

Project Report

(A Microsoft Power BI Data Analyst Professional Certificate Capstone project)

DHANUSHWR K

JUNIOR DATA ANALYST

Table of contents

Introduction: -

This report outlines the process of developing a Power BI dashboard for Tailwind Traders to analyse and visualize sales and profit data. The project involved several key phases, including data preparation, data modelling, DAX calculations, visualization, and dashboard development.

Objective: -

The primary goal of this project is to empower Tailwind Traders with a data-driven decision-making platform. By providing a comprehensive view of global sales and profit performance, the dashboard will enable the company to identify trends, opportunities, and areas for improvement.

Aim: -

Data Preparation

- Data Cleaning: Remove duplicates, inconsistencies, and errors from the Excel data to ensure data accuracy and reliability.
- Data Transformation: Convert data into a suitable format for Power BI analysis, such as standardizing date formats or handling missing values.
- Data Integration: Combine data from different sources (if applicable) into a unified dataset for analysis.

Data Modelling

- Snowflake Schema: Implement a snowflake schema to optimize query performance and improve data organization by creating intermediate dimensions between fact and dimension tables.
- Calendar Table: Create a dedicated calendar table to handle date-related calculations and provide flexibility in time-based analysis.

DAX Calculations

- Key Performance Indicators (KPIs): Develop DAX measures to calculate:
    - Gross Sales: The total revenue generated before deducting expenses.
    - Net Sales: The total revenue generated after deducting discounts, returns, and allowances.
    - Profit Margins: The percentage of revenue remaining after deducting costs.
    - Time-Based Summaries: Analyse sales and profit performance over specific time periods (e.g., monthly, quarterly, yearly).

Visualization

- Interactive Charts: Use charts like line charts, bar charts, and scatter plots to visualize trends and relationships.
- Graphs: Employ graphs like treemaps and sunburst charts to represent hierarchical data and identify patterns.
- Cards: Display key metrics and KPIs in a concise and easy-to-read format.
- Filters and Slicers: Allow users to drill down into specific data segments and customize their analysis.

Dashboard Development

- User-Friendly Design: Create a visually appealing and intuitive dashboard layout.
- Clear Organization: Group related visualizations and metrics to enhance understanding.
- Storytelling: Use storytelling techniques to guide users through the dashboard and highlight key insights.

Subscription and Alerts

- Scheduled Refresh: Configure automatic data refreshes to keep the dashboard up-to-date.
- Subscriptions: Set up email subscriptions to deliver regular updates to stakeholders.
- Alerts: Create alerts to notify users of significant changes or deviations from target KPIs.

By following these steps, Tailwind Traders can develop a powerful Power BI dashboard that provides actionable insights into their global sales and profit performance, driving informed decision-making and improving overall business outcomes.

Prepare & process: -

Part 1: Data Preparation

- **Excel Data Cleaning: Ensured data accuracy and consistency in the Excel sales, purchase and countries data.**

Here's a breakdown of the calculated sales data:

1. **Gross Revenue:** The total revenue generated from each product, calculated by multiplying the Quantity Purchased by the Unit Price.
2. **Total Tax:** The tax amount for each product, calculated by multiplying the Quantity Purchased by the Tax Rate.
3. **Net Revenue:** The actual earnings for each product after deducting taxes, calculated by subtracting Total Tax from Gross Revenue.

No changes required in purchase and countries data

- **Data Integration: Imported the cleaned sales, purchase, countries data into Power BI, defining appropriate data types.**

Here's a breakdown of the configuring data sources:

1. **Sales Data**

- **Data Quality Checks:**
  - Verified 100% validity for Order ID column.
  - Analysed histogram frequency of distinct and unique values in Gross Product Price.
  - Recorded MIN, MAX, and AVERAGE values for Quantity Purchased.

➢ **Data types assigned:**

- **Gross Product Price =** Fixed Decimal Number
- **Tax Per Product =** Fixed Decimal Number
- **Quantity Purchased =** Whole Number
- **Loyalty Points =** Whole Number
- **Stock =** Whole Number
- **Product Category =** Text
- **Rating =** Fixed Decimal Number

2. Purchases Data

- **Data Quality Checks:**
  - Verified 100% validity for Return Status column.
  - Recorded MIN, MAX, and AVERAGE values for Warranty (Months).
- **Data types assigned:**
  - **Purchase ID** = Whole Number
  - **Order ID** = Whole Number
  - **Return Policy (Days)** = Whole Number
  - **Purchase Date** = Date
  - **Warranty (Months)** = Whole Number
  - **Supplier** = Text
  - **Last Visited** = Date
  - **Return Status** = Text
- **Filtered data:** Filtered Return Status column to show only "Not Returned" records.

3. Countries Data

- **No calculations required.**

- **Data types assigned:**
    - **Country ID =** Whole Number
    - **Exchange ID =** Whole Number
    - **Country =** Text

➢ **Historical Currency Exchange Data**

- **Data Transformation: Converted raw string data into a structured Data Frame using Python.**

  Script:

```
import pandas as pd
from io import StringIO

data = """Exchange ID;ExchangeRate;Exchange Currency
1;1;USD
2;0,75;GBP
3;0,85;EUR
4;3,67;AED
5;1,3;AUD"""
df = pd.read_csv(StringIO(data), sep=';')
```

- **Calculated data:**
    - **Exchange Rate:** The exchange rate for each currency relative to USD.

Part 2: <u>Data Modelling</u>

➢ **Snowflake Schema: Designed a snowflake schema to optimize data storage and performance.**

➢ **Relationship Creation: Established relationships between tables in the data model.**

1) **Countries and Exchange Data:** Established a relationship between Countries and Exchange Data, linking them based on the Exchange ID field.

- Cardinality: One to One (1:1).
- Cross Filter Direction: Cross filter direction was set to Both to enable bidirectional filtering between the two tables.

2) **Sales and Countries:** Created a relationship between Sales and Countries, connecting them using the Country ID field.
   - Cardinality: Many to One (*:1).
   - Cross Filter Direction: Cross filter direction was set to Both to enable bidirectional filtering between the two tables.

3) **Purchases and Sales:** Established a relationship between Purchases and Sales, linking them based on the Order ID field.
   - Cardinality: One to One (1:1).
   - Cross Filter Direction: Cross filter direction was set to Both to enable bidirectional filtering between the two tables.

➢ **Calendar and Sales in USD table: Created a dedicated calendar and sales in USD table for temporal analysis.**

1) **Created a Calendar Table**

- Created a new table using DAX code to provide a granular time-based dimension.
- Defined columns for Year, Month Number, Month, Quarter, Weekday, and Day.

DAX code:

```
CalendarTable =
ADDCOLUMNS(
CALENDAR(DATE(2020, 1, 1), DATE(2023, 12, 31)),
"Year", YEAR([Date]),
"Month Number", MONTH([Date]),
"Month", FORMAT([Date], "MMMM"),
"Quarter", QUARTER([Date]),
"Weekday", WEEKDAY([Date]),
"Day", DAY([Date])
)
```

2) Created a Relationship between Calendar and Purchases

- Established a relationship between Calendar and Purchases, joining on the Date field.
    - Cardinality: Many to One (*:1).
    - Cross Filter Direction: Cross filter direction was set to single

3) Created a Sales in USD Calculated Table

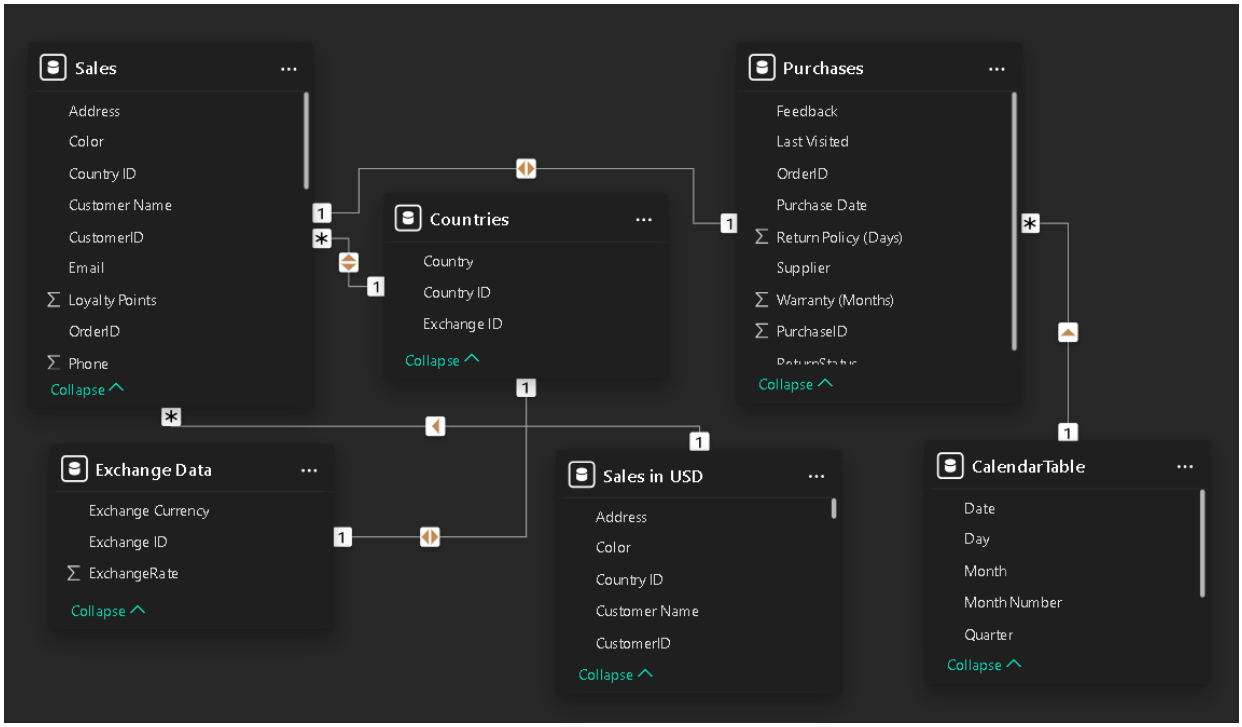
- Created a new table using DAX code to calculate sales data in USD.
- Included columns for Country Name, Exchange Rate, Exchange Currency, Gross Revenue USD, Net Revenue USD, and Total Tax USD.
- Used RELATED functions to retrieve the corresponding Exchange Rate for each sale and calculated the USD values.

DAX code:

```
Sales in USD =
ADDCOLUMNS(
   Sales,
   "Country Name", RELATED(Countries[Country]),
   "Exchange Rate", RELATED('Exchange Data'[Exchange Rate]),
   "Exchange Currency", RELATED('Exchange Data'[Exchange Currency]),
    "Gross Revenue USD", [Gross Revenue] * RELATED('Exchange Data'[Exchange Rate]),
    "Net Revenue USD", [Net Revenue] * RELATED('Exchange Data'[Exchange Rate]),
   "Total Tax USD", [Total Tax] * RELATED('Exchange Data'[Exchange Rate])
)
```

4) Created a Relationship between Sales in USD and Sales

- Established a relationship between Sales in USD and Sales, joining on the Order ID field.
    - Cardinality: Many to One (*:1).
    - Cross Filter Direction: Cross filter direction was set to single



Analyse: -

Part 1: DAX Calculations

- **Key Performance Indicators:** Developed DAX measures to calculate gross sales, net sales, profit margins, and other relevant KPIs.
- **Time-Based Summaries:** Created measures for quarterly, annual, and year-to-date profit analysis.

1) Data Calculated:

- **Yearly Profit Margin:** The percentage of profit earned relative to total revenue for each year.
- **Quarterly Profit:** The total profit generated during each quarter of the year.
- **Year-to-Date Profit:** The cumulative profit earned from the beginning of the year up to the current date.
- **Median Sales:** The middle value of gross sales, representing the typical sales amount.

Specific Calculations:

- **Yearly Profit Margin:** Calculated using the formula: Yearly Profit Margin = (Gross Revenue - Total Net Revenue) / Total Net Revenue
- **Quarterly Profit:** Calculated using the DATESQTD function to filter data for the current quarter and then summing the Yearly Profit measure.
- **Year-to-Date Profit:** Calculated using the TOTALYTD function to filter data from the beginning of the year to the current date and then summing the Yearly Profit measure.
- **Median Sales:** Calculated using the MEDIAN function on the Gross Revenue column.

Currency Conversion: Calculated sales and profit data in US dollars using currency exchange rates.

2) Performance Analysis:

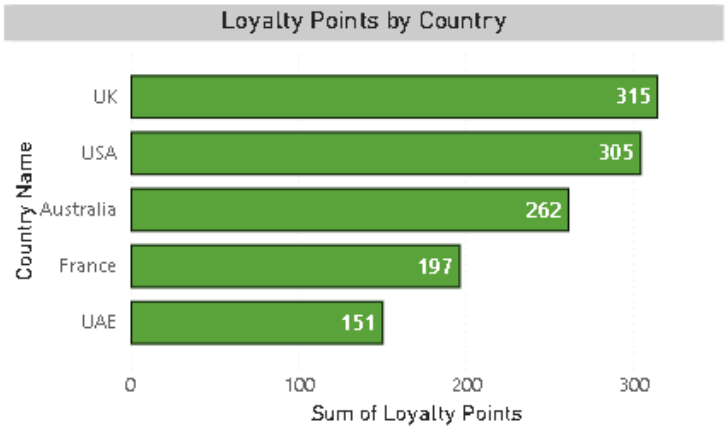
- **Load Times:** Measured the load times of card visuals containing the calculated measures using performance analyser.
- **DAX Query Time:** Recorded the time taken to execute DAX queries for each visual.
- **Slow-Loading Visuals:** Identified visuals with load times exceeding 200ms.

Part 2: Visualization

- **Chart Creation:** Developed various charts (e.g., line charts, bar charts, pie charts) to visualize sales and profit trends.
- **Card and KPI Display:** Used cards and KPIs to highlight key metrics and performance indicators.

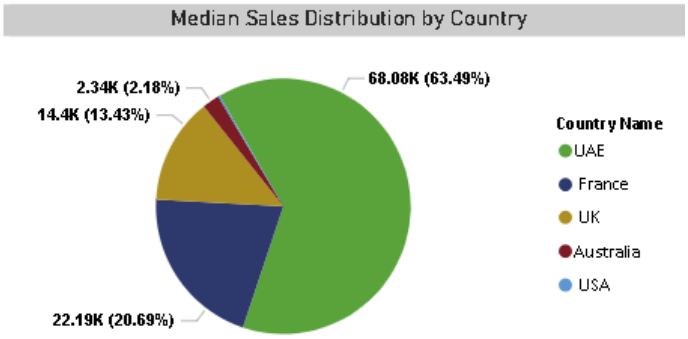
1) Creating the Sales Overview Report

- Created a **New Report** and renamed the Tailwind Traders Report.pbix file to Sales Overview.
- Visualized **Loyalty Points by Country**. Created a bar chart to show loyalty points by country. Configured the chart with country names on the Y-axis and loyalty points on the X-axis.

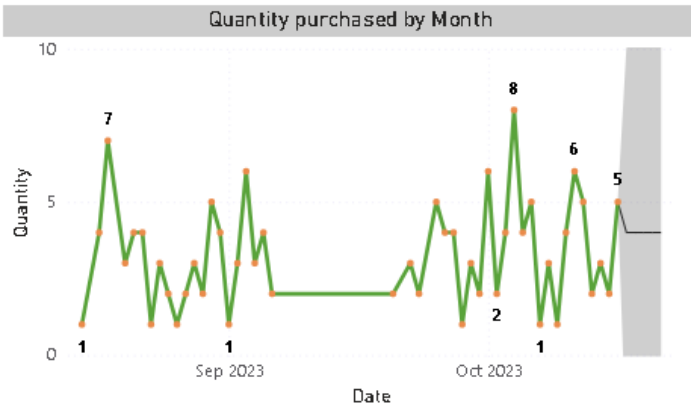


- Visualized **Quantity Sold by Product**. Created a column chart to show quantity sold by product. Configured the chart with product names on the Y-axis and quantity purchased on the X-axis.

- Visualized **Median Sales Distribution by Country**. Created a pie chart to show median sales distribution by country. Configured the chart to display country names and median sales.



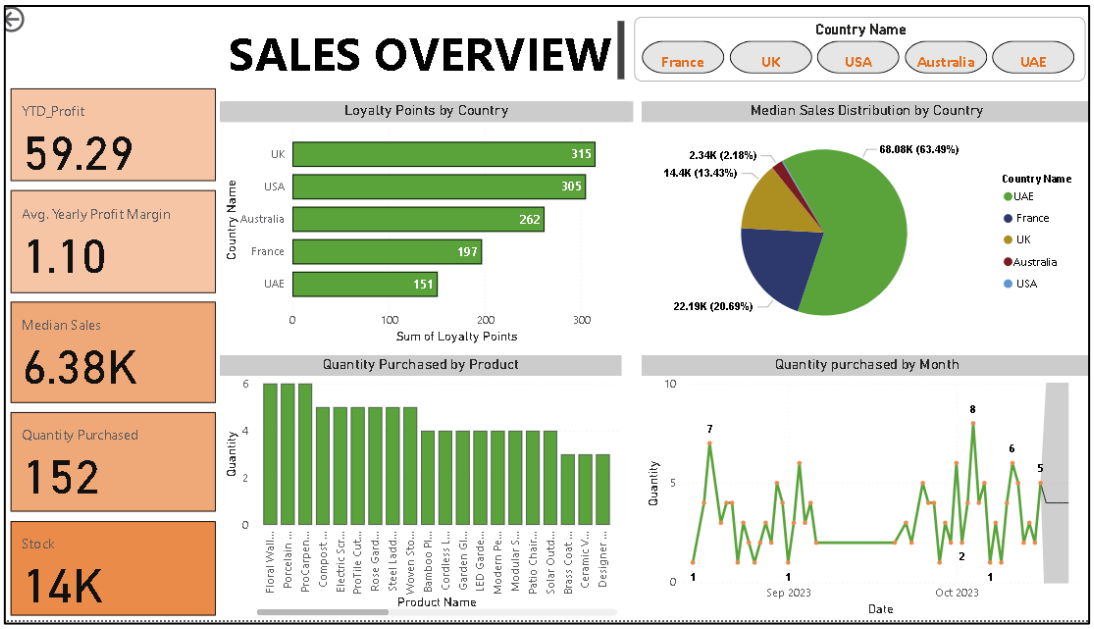
- Visualized **Quantity purchased Over Time**. Created a line chart to show Quantity purchased over time. Configured the chart to display date data on the X-axis and Quantity purchased data on the Y-axis. Added a forecast using the Analytics pane.



- Created Cards for **Key Metrics**. Created cards to display **Stock, Quantity Purchased, and Median Sales**.

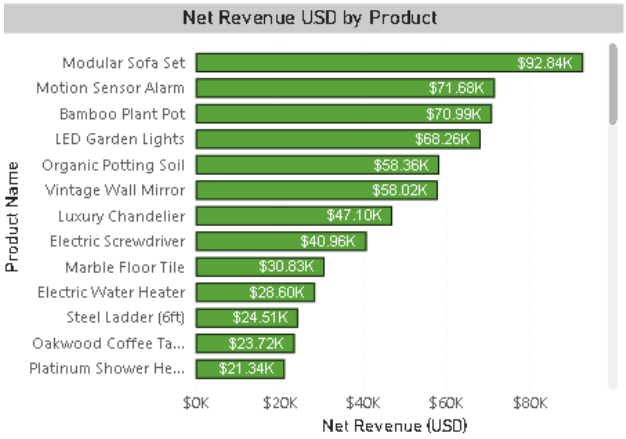
- Added a **Slicer**. Created a slicer to filter data **by Country Name** and saved.

SALES OVERVIEW REPORT:

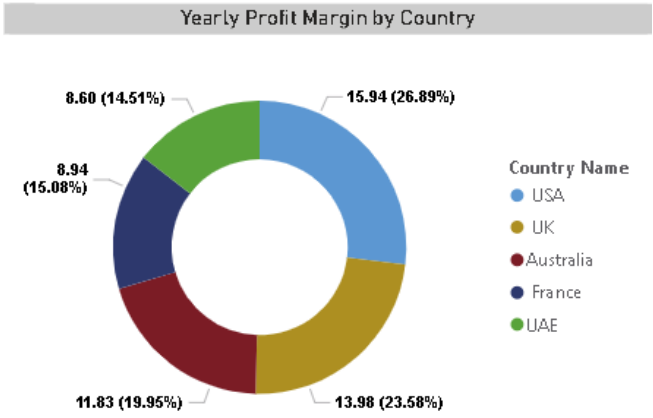


2) Creating the Profit Overview Report

- Created a New Page. Opened the Sales Overview report and **created a new page named Profit Overview.**
- Visualized **Net Revenue by Product**. Created a bar chart to show net revenue by product.

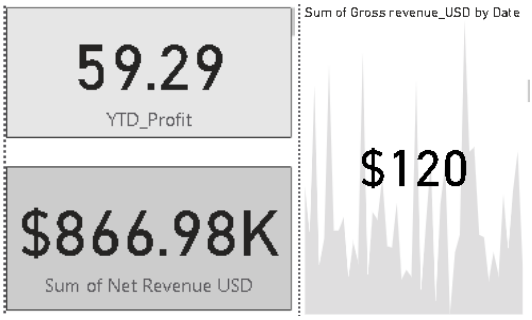


- Visualized **Yearly Profit Margin by Country**. Created a donut chart to show yearly profit margin by country.



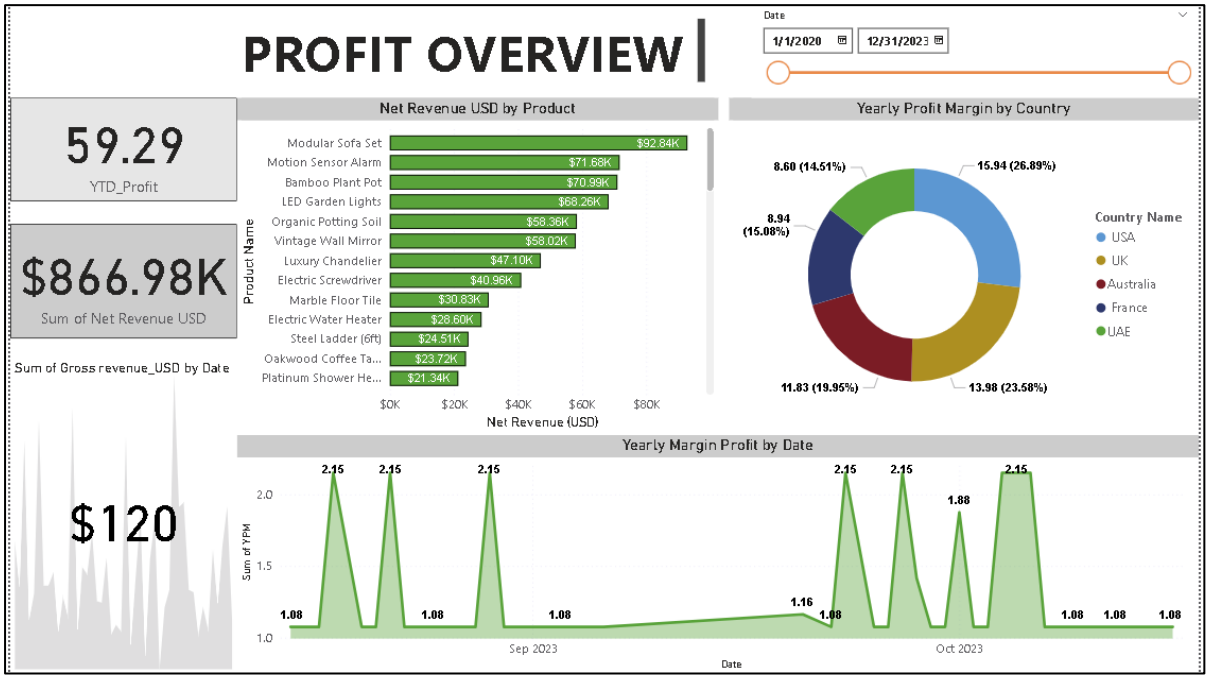
- Visualized **Yearly Profit Margin over Time**. Created an area chart to show yearly profit margin over time.

- Created Cards for **Key Metrics**. Created cards to **display YTD Profit and Net Revenue USD.**
- Created a **KPI** for **Gross Revenue USD**



- Added a **Slicer.** Created a slicer to filter **data by date**, saved and published the report.

PROFIT OVERVIEW REPORT:



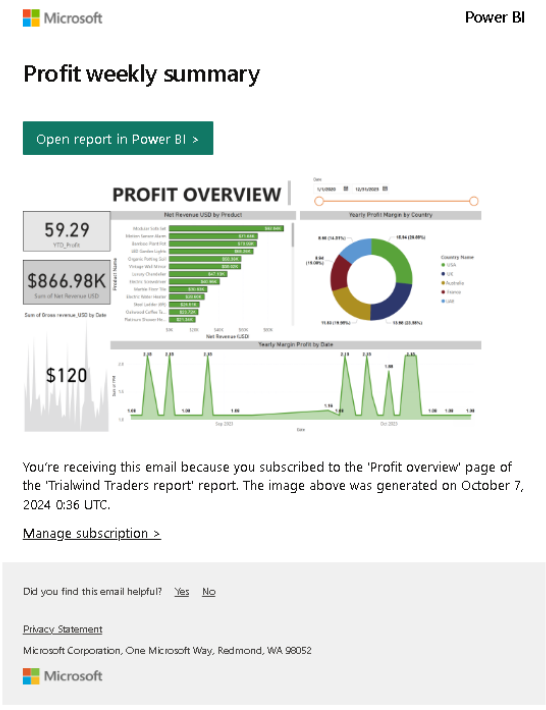
Share: -

Part 1: <u>Dashboard Development</u>

- **Dashboard Layout:** Designed a visually appealing and intuitive dashboard layout.
- **Dashboard Elements:** Incorporated charts, graphs, cards, and filters to provide a comprehensive overview of sales and profit data.
- **Mobile Optimization:** Ensured the dashboard is accessible and responsive on mobile devices.

Part 2: <u>Subscriptions and Alerts</u>

- **Subscription Configuration:** Set up subscriptions to deliver scheduled reports to stakeholders.
- **Alert Creation:** Defined alerts to notify users of significant changes in key metrics.
  - Created an alert for Gross Revenue USD falling below $400.
  - Created a weekly subscription for the Sales Overview report. Set the subscription to send on Mondays at 5:00 AM.
  - Created a weekly subscription for the Profit Overview report. Set the subscription to send on Mondays, Wednesdays, and Fridays at 6:00 AM.



Conclusion: -

The Tailwind Traders dashboard successfully addresses the company's need for actionable insights into global sales and profit performance. By providing a centralized platform for data analysis and visualization, the dashboard empowers decision-makers to identify trends, optimize operations, and drive business growth.