

CHAPTER 1

INTRODUCTION

The primary objective of this project is to develop a Power BI dashboard that delivers real-time sales insights, enhancing the decision-making process for sales directors and regional managers. By integrating data from a MySQL database into Power BI, we aim to:

- Reduce Costs:** Achieve a 10% reduction in costs related to manual data merging and processing.
- Increase Efficiency:** Streamline sales tracking and reporting processes.
- Improve Sales Performance:** Achieve a 5% improvement in sales within three months post-implementation.

Power BI is a business analytics tool developed by Microsoft, designed to provide interactive visualizations and business intelligence capabilities. It enables users to connect to multiple data sources, transform raw data into meaningful insights, and share these insights across the organization.

Key Features of Power BI Data Connectivity: Power BI supports a wide range of data sources, including SQL databases, Excel files, cloud services, and more. This allows for seamless integration and consolidation of diverse data sets.

Data Transformation: The Power Query Editor within Power BI allows for robust data cleaning and transformation, ensuring data accuracy and consistency before analysis.

Interactive Visualizations: Power BI offers a variety of visualization options, including charts, graphs, and maps, which can be customized and interacted with to explore data from different perspectives.

Real-Time Analytics: With Power BI, users can create dashboards that update in real-time, providing up-to-date insights that are crucial for timely decision-making.

Collaboration and Sharing: Dashboards and reports can be easily shared with team members, facilitating collaborative analysis and discussion.

MySQL Workbench MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. It provides data modelling, SQL development, and comprehensive administration tools for server configuration, user administration, and backup.

Database Design: MySQL Workbench facilitates the design and modelling of databases through an intuitive graphical interface.

SQL Development: It offers an integrated SQL editor for writing and executing queries, aiding in data extraction and transformation.

Server Administration: The tool includes features for server management and administration, such as user account management and performance monitoring.

CHAPTER 2

PROPOSED SYSTEM

To address the challenges faced by the sales department in tracking sales and gaining business insights, we propose the implementation of a comprehensive sales insights system using Power BI. This system will integrate data from various sources, transform it for analysis, and present it through interactive dashboards. Below are the key components and functionalities of the proposed systems:

1. Data Integration System

Objective : To consolidate sales data from multiple sources into a unified database for streamlined access and analysis.

Components:

- MySQL Database: Centralized storage for all sales-related data.
- Data Import Scripts: Automated scripts to import data from various sources (e.g., CSV files, ERP systems) into the MySQL database.
- Data Integrity Checks: Procedures to ensure the accuracy and consistency of the imported data.

Benefits:

- Ensures all sales data is stored in a single, accessible location.
- Facilitates data consistency and accuracy.
- Reduces manual data entry and associated errors.

2. Data Transformation System

Objective: To clean and transform raw data into a format suitable for analysis.

Components:

- Power Query Editor: Tool within Power BI used for data cleaning and transformation.
- Data Cleaning Procedures: Steps to remove duplicates, handle missing values, and correct errors in the data.
- Data Normalization: Processes to standardize data formats, such as currency conversions and date formatting.

Benefits:

- Ensures data quality and reliability.

- Prepares data for efficient analysis and visualization.
- Automates repetitive data cleaning tasks, saving time and effort.

3. Data Modeling System

Objective: To establish logical relationships between different data entities for comprehensive analysis.

Components:

- Star Schema Design: A data modeling approach where a central fact table (e.g., sales transactions) is connected to dimension tables (e.g., customers, products, regions).
- Relationship Mapping: Defining and establishing relationships between tables in Power BI to enable multidimensional analysis.

Benefits:

- Simplifies complex data structures, making them easier to analyze.
- Enhances the ability to generate insights across various dimensions (e.g., time, geography, product).
- Facilitates efficient query performance in Power BI.

4. Visualization and Dashboard System

Objective: To create interactive dashboards that provide real-time sales insights for decision-making.

Components:

- Power BI Dashboards: Interactive visualizations, including charts, graphs, and tables, to display key sales metrics.
- Custom Visualizations: Tailored visual elements to meet specific business requirements and preferences.
- Filters and Slicers: Tools to allow users to dynamically explore data by applying various filters (e.g., time period, region, product).

Benefits:

- Provides a clear and concise view of sales performance.
- Enables users to explore data and gain insights interactively.
- Supports real-time data updates, ensuring timely information is always available.

5. Stakeholder Feedback and Iteration System

Objective: To continuously improve the dashboard based on stakeholder feedback to ensure it meets business needs effectively.

Components:

- Feedback Collection Mechanism : Structured process for gathering feedback from stakeholders, such as surveys or video comments.
- Feedback Analysis and Prioritization : Methods to categorize and prioritize feedback for implementation.
- Iteration and Enhancement : Regular updates to the dashboard based on feedback, ensuring it evolves to meet changing business requirements.

Benefits:

- Ensures the dashboard remains relevant and useful to stakeholders.
- Involves stakeholders in the development process, increasing their engagement and satisfaction.
- Facilitates continuous improvement and adaptation to new business challenges.

6. Deployment and Access System

Objective: To deploy the final dashboard and ensure it is accessible to all relevant stakeholders.

Components:

- Power BI Service : Platform for publishing and sharing Power BI reports and dashboards.
- Mobile Access : Configuration to allow access to the dashboard via mobile devices for on-the-go insights.
- User Training and Support : Providing training sessions and support resources to help users effectively utilize the dashboard.

Benefits:

- Ensures broad access to critical sales insights across the organization.
- Facilitates timely and informed decision-making.
- Provides flexibility for users to access insights from anywhere.

By implementing these proposed systems, we aim to create a robust and insightful sales insights platform that enhances data-driven decision-making, improves sales performance, and supports the overall business objectives of the organization.

CHAPTER 3

SYSTEM DESIGN

Designing a system that integrates Power BI for sales insights with MySQL Workbench involves several key components. Here's a step-by-step overview -

1. Understanding Requirements

- Identify the specific sales metrics and insights needed.
- Determine the data sources and data integration requirements.
- Establish performance and scalability requirements.

2. Database Design with MySQL Workbench

- Schema Design: Design the database schema in MySQL Workbench to store sales data. This includes tables for sales transactions, products, customers, sales staff, etc.
- Relationships: Define relationships between tables (e.g., foreign keys).
- Indexes: Create indexes to improve query performance.
- Stored Procedures/Views: Write stored procedures and views for complex queries that Power BI can call directly.

3. Data Extraction, Transformation, and Loading (ETL)

- Using ETL tools and scripts like SQL Workbench and Power Query to extract data from various sources, transform it into the desired format, and load it into the MySQL database.
- Ensuring data is cleaned and pre-processed to maintain quality.

4. Connecting Power BI to MySQL

- Direct Query or Import Mode: For this project, we have directly imported the data from SQL Database by establishing local host connection. This ensures that the Dashboard remains updated.
- Data Gateway: Set up a data gateway if the MySQL database is on-premises to allow Power BI to access it.
- Credentials: Configure the connection settings and credentials in Power BI to connect to the MySQL database.

5. Data Modeling in Power BI

- **Tables and Relationships:** Importing the relevant tables and defining relationships within Power BI table view.
- **Measures and Calculations:** We have created new measures and calculated columns for custom metrics and KPIs using DAX language.
- **Hierarchies:** Set up hierarchies for easier navigation and drilling down into data.

6. Report and Dashboard Design in Power BI

- **Visualization:** We have created two Dashboards – one related to Sales and the other related to Key Insights. The interactive visuals such as charts, tables, and maps to represent sales data allow Managers to take informed decisions.
- **Filters and Slicers:** Implementing filters and slicers to allow users to drill down into specific data segments.
- **Themes and Layout:** Designing a user-friendly and visually appealing layout, applying consistent themes and branding.

7. Performance Optimization

- **Query Performance:** Optimizing SQL queries and DAX calculations for performance.
- **Data Refresh:** Scheduling data refreshes in Power BI to ensure data is up-to-date without overloading the system.

8. Testing and Validation

- **Data Accuracy:** Validating the accuracy of the data being displayed in Power BI.
- **Performance Testing:** Testing the performance of the reports and dashboards under different load conditions.

9. Deployment and Maintenance

- **Deployment:** Deploying the Power BI reports and dashboards to the Power BI Service.
- **Monitoring:** Setting up monitoring for data refreshes and report performance.
- **Continuous Improvement:** Gathering user feedback and continuously improve the system.

System Architecture

1. Data Source:

- We have obtained the open-source sales data released by the company AtliQ Hardware which contains data related to sales from 2016 to 2020.

2. ETL Process:

- Extract data from sources.
- Transform data using an ETL tools like Workbench and Power Query.
- Load data into our local MySQL database.

3. MySQL Database:

- Tables for sales transactions, products, customers, regions and markets, etc.
- Stored procedures and views for complex queries.

4. Power BI:

- Connect to MySQL database.
- Data modeling and DAX calculations.
- Creating reports and dashboards.
- Publish to Power BI Service.

5. End Users:

- Access Power BI dashboards and reports via web or Power BI desktop app.
- Apply filters and slicers to explore data.

This approach ensures a comprehensive and scalable system for generating sales insights using Power BI and MySQL Workbench.

CHAPTER 4

IMPLEMENTATION

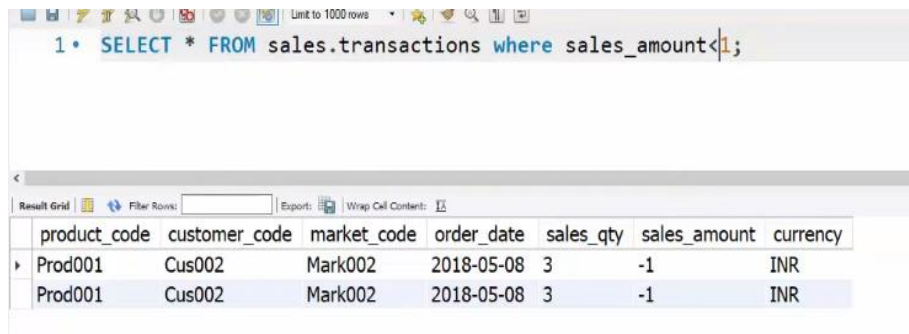
Implementation Steps for the Sales Insights Dashboard

1. Data Integration

The initial step involved integrating data from MySQL, the primary database for AtliQ Hardware. We established secure connections to extract sales data, ensuring data integrity and consistency. Using SQL queries, we gathered comprehensive sales records, including details on products, regions, clients, and sales transactions. Below is an example query used to extract sales data -

```
SELECT sales.id, sales.date, products.product_name, regions.region_name,
clients.client_name, sales.quantity, sales.total_amount FROM sales
JOIN products ON sales.product_id = products.id
JOIN regions ON sales.region_id = regions.id
JOIN clients ON sales.client_id = clients.id;
```

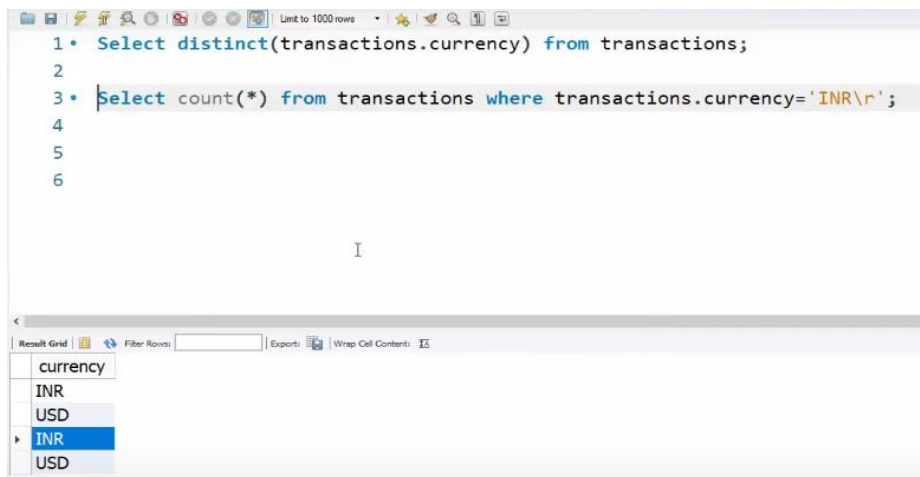
The following images show few queries that were run on MYSQL



```
1 • SELECT * FROM sales.transactions where sales_amount<1;
```

product_code	customer_code	market_code	order_date	sales_qty	sales_amount	currency
Prod001	Cus002	Mark002	2018-05-08	3	-1	INR
Prod001	Cus002	Mark002	2018-05-08	3	-1	INR

Figure 1



```
1 • Select distinct(transactions.currency) from transactions;
2
3 • Select count(*) from transactions where transactions.currency='INR\r';
4
5
6
```

currency
INR
USD
INR
USD

Figure 2

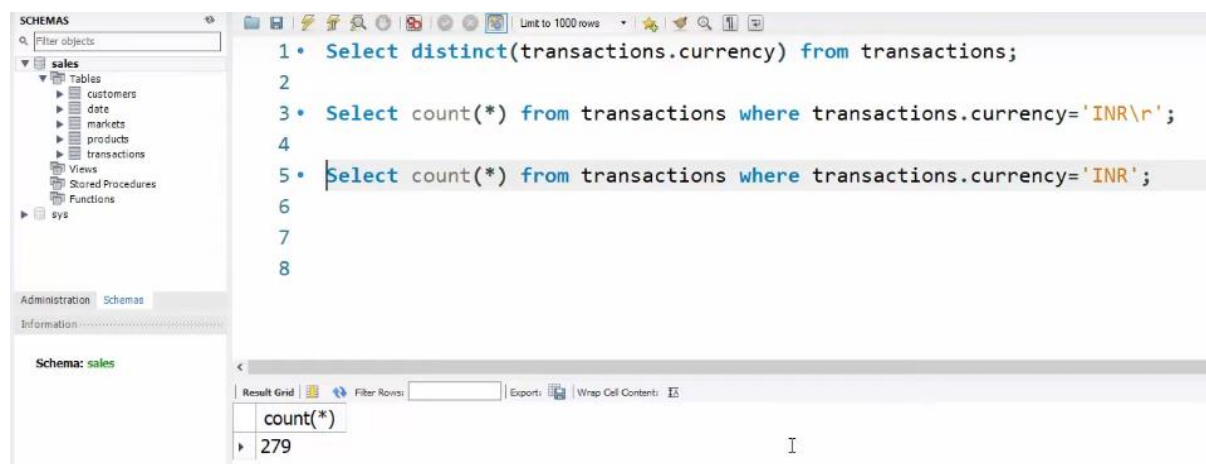


Figure 3

2. Data Cleaning in MySQL Workbench

The data cleaning process in SQL was pivotal in ensuring the accuracy and reliability of the dataset used for the sales insights dashboard. This phase involved writing SQL queries to identify and rectify various data anomalies. We addressed missing values by using techniques such as imputation and default value assignment. Duplicate records were identified using `DISTINCT` and `GROUP BY` clauses and were subsequently removed to prevent data redundancy. Inconsistencies in data formats, particularly in date and numerical fields, were standardized using SQL functions. Additionally, data validation checks were performed to ensure the integrity of the dataset, such as verifying that sales figures were non-negative and that product IDs matched entries in the product master table.

The query, `SELECT count(*) FROM transactions WHERE transactions.currency='INR\r';` counted the number of transactions where the currency code was incorrectly recorded as 'INR\r'. This step highlighted the presence of an anomaly where the currency code included an unexpected carriage return character. The query, `SELECT count(*) FROM transactions WHERE transactions.currency='INR';`, counted the transactions correctly recorded with the currency code 'INR'. This step ensured that the correctly formatted entries were identified separately.

By identifying these anomalies, we could perform necessary corrections. The carriage return character (`\r`) was removed from the currency codes, ensuring uniformity. Furthermore, all currency values were standardized and converted to INR (Indian Rupees) using exchange rates provided in the dataset or through a standard conversion formula. This ensured consistency

across financial data and facilitated easier analysis of sales figures. By thoroughly cleaning the data in SQL before importing it into Power BI, we established a strong foundation for accurate and meaningful analysis, ultimately enhancing the reliability of the insights derived from the dashboard.

3. Data Transformation using Power Query Editor

Once we are connected to the Primary Database, we can see a preview of the available tables. Click "Load" to import the data into Power BI or "Transform Data" to open the Power Query Editor for data transformation.

We established relationships between different tables (e.g., connecting customer_code in the customer table to customer_code in the sales_transaction table) and ensured that proper relationships are set to form a star schema, where we have fact tables (e.g., sales_transaction) and dimension tables (e.g., customer, product, market).

We created custom columns using Power Query's M language. This allows for a wide range of transformations. For example, we added a column for profit margin with the below code -

$$[\text{Profit Margin}] = ([\text{Sales Amount}] - [\text{Cost}]) / [\text{Sales Amount}]$$

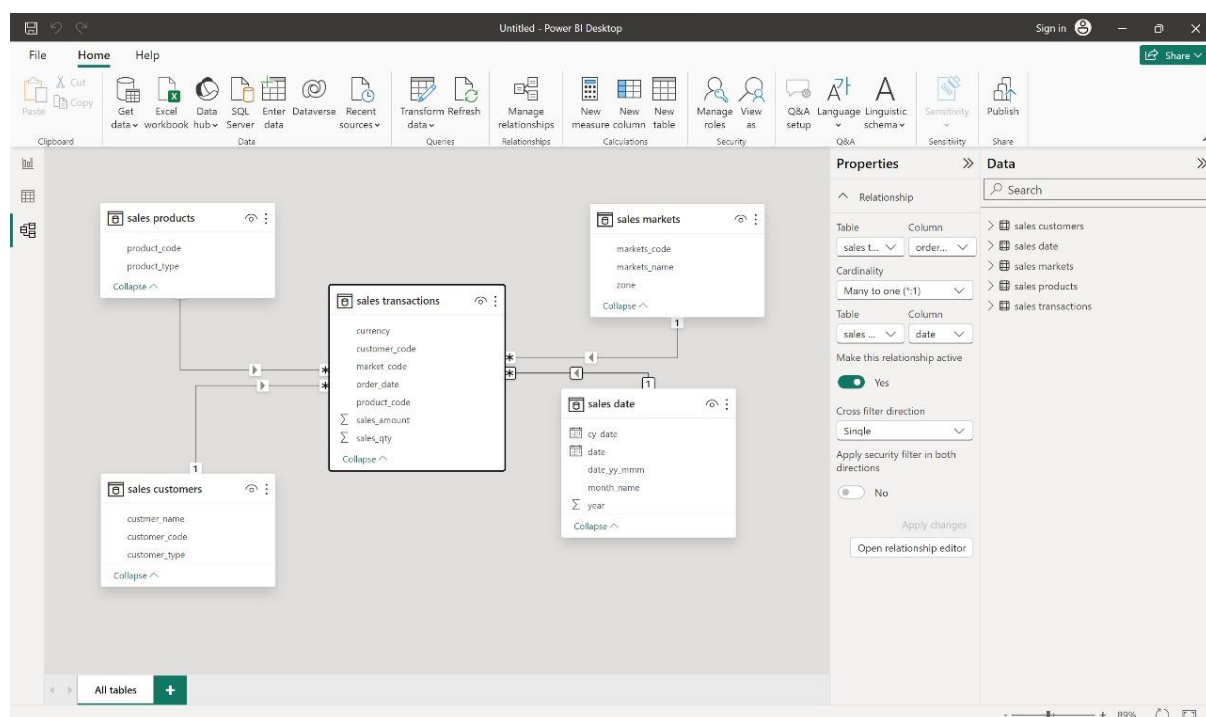


Figure 4: Relationships between fact table and dimension tables

We also created many measures in Sales Database, for example Revenue and Sales Quantity. Measures are calculations used in aggregating data, useful for dynamic analysis. For example Calculating total sales -

$$\text{Total Sales} = \text{SUM}('Sales'[\text{Sales Amount}])$$

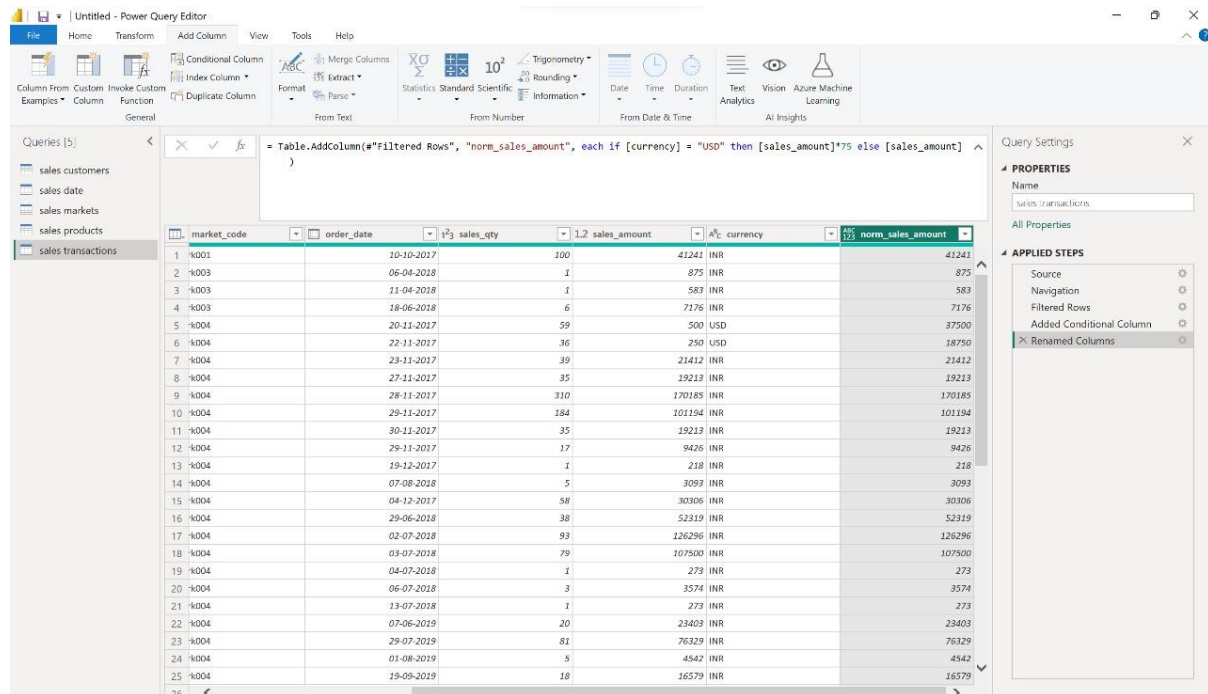


Figure 5: New Measure called Revenue

4. Dashboard Design

The design of the Power BI dashboard focused on creating an intuitive and interactive user interface. We aimed to present data in a visually appealing and easily understandable manner. We have created 2 Dashboards one for Sales Insights and the other for Key Insights related to profit. Key visualizations included Bar, Line, Pie, Column, Scattered, Ribbon and Table charts, providing quick insights into overall sales performance and targets.

Interactive elements such as slicers, filters, and drill-through capabilities were incorporated to allow users to customize their views and delve deeper into specific data subsets.



Figure 6: Overview of Sales Dashboard

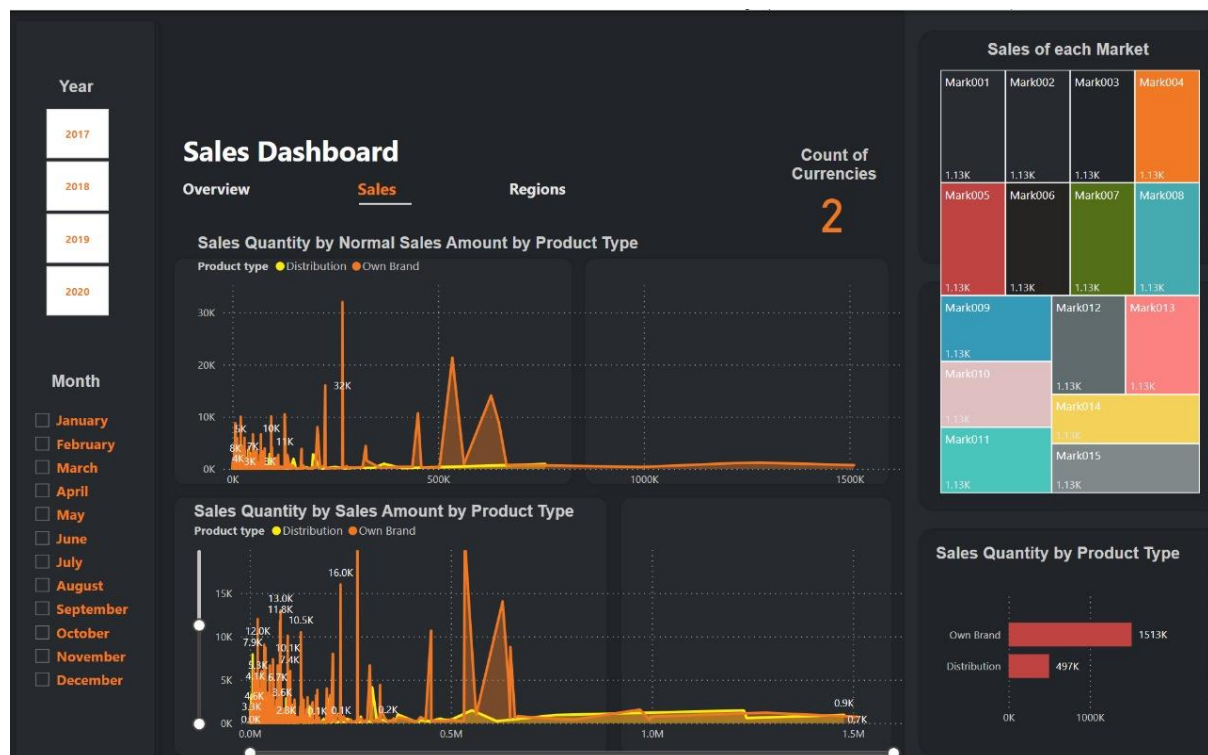


Figure 7: Sales Insights page

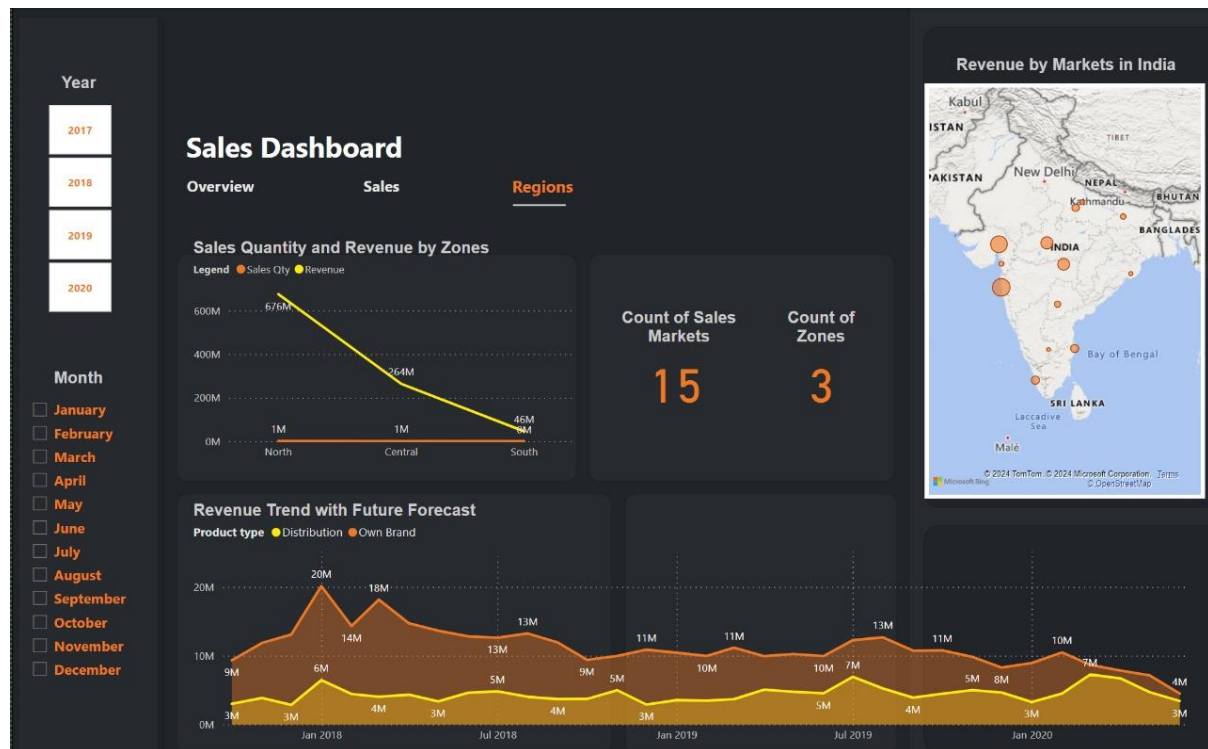


Figure 8: Regions wise analysis of Sales Dashboard



Figure 9: Overview of Key Insights

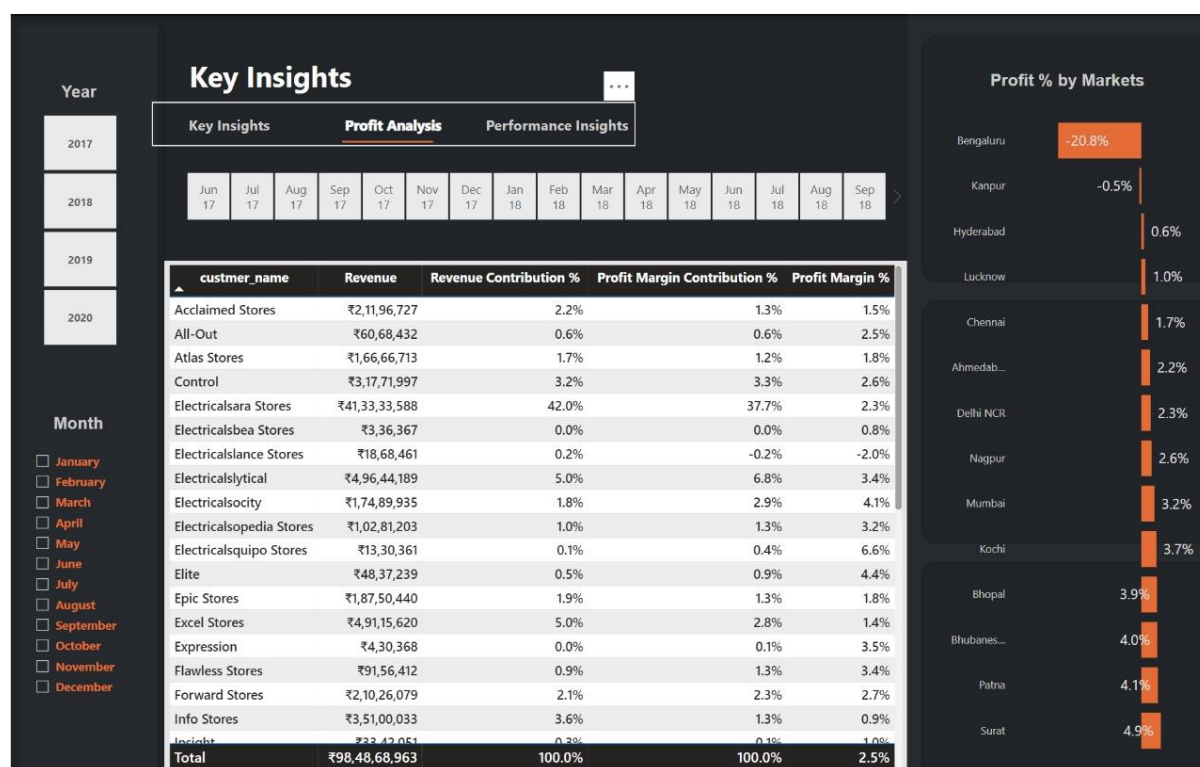


Figure 10: Profit analysis



Figure 11: Performance Insights

Project Margin is a critical metric in sales analytics and business performance management. In a Power BI dashboard, it serves as a key indicator of profitability and efficiency. By analyzing the margin, you can identify which projects or products are generating more profit and which are underperforming.

High Margin: Indicates that a project or sale is highly profitable.

Low Margin: Suggests that costs might be too high or pricing strategies might need adjustment.

5. Deployment and Access Configuration

With the final version of the dashboard ready, we proceeded to deploy it on the Power BI service. This step involved publishing the report to the web, setting up data refresh schedules to ensure real-time updates, and configuring access permissions. We ensured that the dashboard was accessible to all relevant stakeholders, both through desktop and mobile devices, providing flexibility and convenience in data access. The deployment process also included setting up security measures to protect sensitive data.

While deploying the final version of the dashboard on the Power BI service, we encountered a significant limitation due to the absence of a work email. Without a work email, we were unable to access advanced features such as creating Power BI Apps, setting up automated data refresh schedules, and implementing row-level security. As a result, we had to rely on a Power BI Pro trial for some features, use basic sharing options, and manually manage data refreshes to ensure that the dashboard remained up-to-date and accessible to all relevant stakeholders. Despite these challenges, we successfully published the dashboard and provided access through both desktop and mobile devices, along with implementing basic security measures to protect sensitive data.

CONCLUSION

The implementation of a Sales and Profit Insights dashboard using Power BI addresses the significant challenges faced by the sales department in tracking sales and gaining business insights. By integrating data from MySQL, transforming it for analysis, and presenting it through interactive dashboards, we have created a powerful tool that provides accurate, real-time sales data. This enables sales directors and regional managers to make informed decisions and adapt to market changes swiftly.

The proposed systems for data integration, transformation, modeling, visualization, stakeholder feedback, and deployment ensure a comprehensive approach to building and maintaining the dashboard. The iterative process of incorporating stakeholder feedback ensures the dashboard remains relevant and effective in meeting business needs. Based on the insights from our sales dashboard, we have identified several key factors that can help us boost our profits and revitalize our business:

1. **Product Quality and Discounts:** Ensure that our products meet high-quality standards to attract and retain customers. Implement strategic discount offers to entice new customers and encourage repeat purchases.
2. **Promotion and Advertising:** Invest in targeted marketing campaigns to increase brand awareness and reach a wider audience.
3. **Healthy Communication:** Foster strong relationships and clear communication channels between management and suppliers to ensure smooth operations.
4. **Product Attraction (Packaging) and Market Identification:** Enhance product packaging to make our offerings more appealing and stand out on the shelves.

By focusing on these areas, we can improve our sales performance and work towards sustainable business growth. This Dashboard could help in increasing the revenue at least by 7% in the next quarter.

Overall, the project aims to reduce costs related to manual data merging, increase efficiency, and improve sales performance, demonstrating the value of data-driven decision-making in a corporate setting.

FUTURE ENHANCEMENTS

While the current implementation provides a robust foundation for sales insights, there are several areas for future enhancement to further improve the dashboard's functionality and effectiveness:

1. Advanced Analytics

Objective: To incorporate advanced analytics and predictive modeling capabilities into the dashboard.

Enhancements:

- Predictive Analytics : Implement machine learning models to forecast sales trends and identify potential market opportunities.
- Anomaly Detection : Use advanced algorithms to detect unusual patterns or outliers in sales data, providing early warnings for potential issues.
- Customer Segmentation : Analyze customer behavior and segment them into distinct groups for targeted marketing and sales strategies.

2. Enhanced Data Integration

Objective : To expand the range of data sources integrated into the dashboard for a more comprehensive view of the business.

Enhancements :

- Third-Party Data Sources : Integrate external data sources such as market research reports, social media data, and competitor analysis for a broader context.
- ERP and CRM Integration : Connect to enterprise resource planning (ERP) and customer relationship management (CRM) systems for a holistic view of sales and customer interactions.

3. Improved User Experience

Objective : To enhance the user interface and experience of the dashboard, making it more intuitive and user-friendly.

Enhancements :

- Customizable Dashboards : Allow users to customize their dashboard views based on their specific needs and preferences.
- Enhanced Interactivity : Introduce more interactive elements such as drill-down capabilities, dynamic filters, and clickable elements for deeper data exploration.

- Performance Optimization : Optimize the dashboard's performance to ensure fast loading times and smooth interactions, even with large datasets.

4. Automation and Alerts

Objective : To automate routine tasks and set up alerts for critical events to ensure timely action.

Enhancements :

- Automated Reporting : Schedule and automate regular reports to be generated and distributed to stakeholders.
- Real-Time Alerts : Set up real-time alerts for key metrics, such as sudden drops in sales or exceeding sales targets, ensuring prompt action is taken.

5. Comprehensive Training and Support

Objective : To provide ongoing training and support for users to maximize the value of the dashboard.

Enhancements :

- Training Programs : Develop comprehensive training programs and resources, including tutorials, webinars, and documentation.
- User Support : Establish a dedicated support team to assist users with any issues or questions they may have.

6. Scalability and Expansion

Objective : To ensure the dashboard can scale with the growing needs of the organization and expand to other business functions.

Enhancements :

- Scalable Architecture : Design the system architecture to handle increasing data volumes and user loads.
- Cross-Functional Dashboards : Expand the use of Power BI dashboards to other business functions such as finance, operations, and HR for a unified data analytics platform.

By implementing these future enhancements, the sales insights dashboard will continue to evolve, providing even greater value and supporting the organization's strategic goals.

REFERENCES

Sources of tools, resources, and methodologies used during implementation are given as follows –

- 1) <https://codebasics.io/>
- 2) <https://www.mysql.com/>
- 3) <https://www.w3schools.com/sql/>
- 4) <https://www.microsoft.com/en-us/power-platform/products/power-bi>
- 5) <https://learn.microsoft.com/en-us/power-bi/>
- 6) https://www.tutorialspoint.com/power_bi/power_bi_quick_guide.htm
- 7) <https://learn.microsoft.com/en-us/dax/>