Sales360 Analysis.
Case Study

Problem Statement:

This case study project focuses on the data engineering aspects of Geocart, an e-commerce ordering and delivery application. The objective is to enhance the shopping experience by utilizing transactional data and analyse it to get better insights. The project aims to design and implement a robust data pipeline that processes and analyzes a relational dataset of customer orders. This dataset, containing anonymized information from over a million orders, will be used to predict user behaviour such as repeat purchases, new product trials, and items added to the cart during a session. The primary goal is to create an efficient and scalable data engineering solution that supports accurate and real-time model predictions to optimize the user's shopping journey. The project will involve data ingestion, transformation, storage, and integration with machine learning models, ensuring a seamless and personalized shopping experience for Geocart users.

What is Expected?

In the initial step of the exploratory analysis, I have generated a document that outlines the outputs of the preliminary data exploration. This document highlights key distributions, and patterns in the data. It also includes a comprehensive list of potential issues and data anomalies that require further investigation and follow-up. These issues range from missing values and outliers to inconsistencies in data entries. The descriptive analysis section of the document emphasizes important outcomes and findings from the data, such as trends in campaign performance metrics, user engagement patterns, and conversion rates.

Moving forward, the next level of analysis will involve a detailed examination of classifying successful and unsuccessful campaigns. I plan to employ various analytical techniques, to identify factors that contribute to campaign success. This analysis will encompass different methods and reports. Through these methods, I aim to uncover insights into the characteristics of successful campaigns and the underlying patterns that differentiate them from unsuccessful ones. The inferences drawn from this comprehensive analysis will provide actionable insights and guide strategic decisions for optimizing future campaign strategies.

1. **Monthly Pricing Summary:** Generate a monthly pricing summary report for all line items shipped within the last 60-120 days. Group the data by RETURNFLAG and LINESTATUS, calculating the totals for extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. Include a count of line items in each group. Present the results in ascending order of RETURNFLAG and LINESTATUS.

2. **Quarterly Pricing Analysis:** Perform a quarterly analysis on pricing data. Calculate the same aggregates as mentioned in the requirements for line items shipped within the last 60-120 days. Group the data by RETURNFLAG and LINESTATUS and display the results in ascending order of these attributes.

3. **Yearly Price Trends:** Analyze yearly pricing trends for line items shipped within the specified date range. Group the data by RETURNFLAG and LINESTATUS, and calculate the total extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. Include the count of line items for each group and present the results sorted by RETURNFLAG and LINESTATUS.

4. **Product Return Analysis:** Focus on analyzing returned products' pricing information within the 60–120-day window. Group the data by RETURNFLAG and LINESTATUS, and calculate the aggregates: extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. Provide the count of line items in each group and sort the results by RETURNFLAG and LINESTATUS.

5. **Discount Effectiveness Report:** Investigate the effectiveness of discounts online items shipped within the specified timeframe. Group the data by RETURNFLAG and LINESTATUS, and compute the aggregates: extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. Include the count of line items for each group and present the results in ascending order of RETURNFLAG and LINESTATUS.

6. **Line-Item Performance Comparison:** Compare the performance of line items shipped within the given date range. Group the data by RETURNFLAG and LINESTATUS, calculating the same aggregates for extended price, discounted extended price, discounted extended price plus tax, average quantity, average extended price, and average discount. Include the count of line items in each group and display the results sorted by RETURNFLAG and LINESTATUS. This analysis can provide insights into the performance of different product categories or order statuses.

Sales360 Analysis.
Case Study

Data Dictionary:
https://github.com/manojkumarsingh77/Shell2023/blob/main/Sales360/DataDictionary/Sales360_DataDictionary.pdf

Data Sets:
https://github.com/manojkumarsingh77/Shell2023/blob/main/Sales360/DataSet/Sales_360.zip

Case Study Execution Plan:

- The execution of each Case Study will involve a group of 4 or 5 members, with each member assigned specific tasks to align with the project's objectives.
- Each group member will work concurrently on their designated tasks, ensuring parallel progress, and the integration of individual contributions will occur during the Final Stage of the project.
- On the Final day, the completed Case Study will be presented to the Shell Subject Matter Experts (SME) and UNext Mentors, providing an opportunity to showcase the project's outcomes and achievements.
- The entire project development process will be implemented using a Continuous Integration/Continuous Deployment (CI/CD) pipeline. This approach ensures seamless integration of code changes, automated testing, and efficient deployment, promoting collaboration and efficiency throughout the project lifecycle.

Technicalities:

In order to address the given problem statement, we will adhere to a standard data pipeline pattern. This structured approach will ensure a systematic and efficient workflow for data processing and transformation.

The data pipeline will consist of the following key stages:

- Data Ingestion.
- Data Processing.
- Data Storage.
- Data Visualization and Reporting.

Data Layers:

As part of a structured data storage approach, you will implement measures to ensure efficient data organization and management. The data will be divided into separate parent folders, one for each team, with sub-folders for **RAW**, **STG (Staging),** and **CURATED** data:

Parent Folders: Each team involved in the project will have its dedicated parent folder to manage their data processing activities. This ensures data isolation and promotes collaboration within the team.

RAW Sub-folder: The RAW sub-folder within each team's parent folder will be used to store the raw and unprocessed data acquired from various sources. This includes the data ingested through Azure Data Factory or any other data ingestion mechanism.

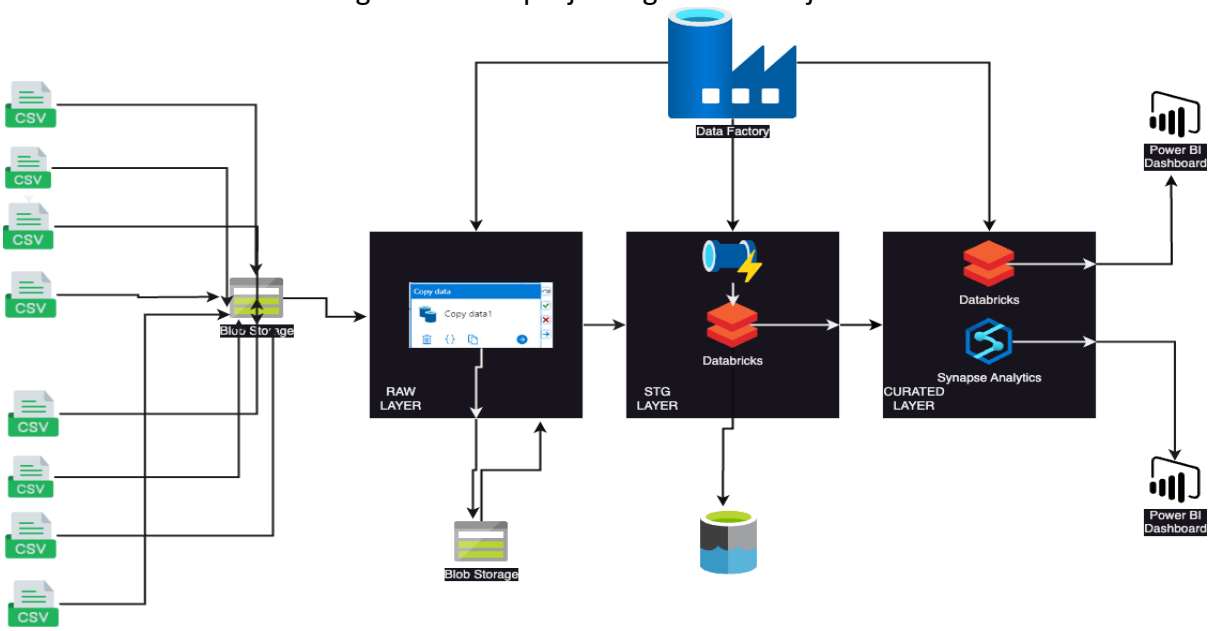
STG (Staging) Sub-folder: The STG sub-folder will serve as an intermediate storage location where data from the RAW sub-folder is transformed and prepared for further processing. This staging step ensures data quality and consistency before moving it to the CURATED sub-folder.

CURATED Sub-folder: The CURATED sub-folder will hold the processed and curated data ready for visualization and analysis. This data is transformed, cleansed, and enriched to meet specific business requirements.

Sales360 Analysis.
Case Study

Reference architecture diagram:

The provided architecture diagram serves as a foundational reference for each team to envision their own version while building upon it. This diagram presents a clear and structured visualization of the system's components and their interactions. Each team is tasked with developing their own iteration of the architecture diagram, using the provided sample as a foundation. This approach fosters creativity and empowers teams to tailor the solution to meet specific requirements and address unique challenges. By building upon the initial reference, teams can explore diverse design choices and leverage individual expertise, resulting in a comprehensive and adaptable solution. This collaborative process ensures a successful outcome that aligns with the project's goals and objectives.



Activity Breakdown:

In the case study, data engineers will perform data ingestion and cleansing activities to ensure data quality and integrity. They will create a reusable and secured connection for data ingestion and handle tasks like removing duplicate records, handling missing values through imputation techniques, and correcting data anomalies.

For ETL and analysis, data engineers will filter out irrelevant or incomplete data, aggregate data to calculate summary statistics, transform data types and create derived columns, perform data joining based on common keys, and apply data partitioning for improved query performance. They will also conduct data deduplication and implement validation checks to ensure data quality and adherence to business rules.

The Case Study is divided into two parallel streams, each handled by separate teams:

i. **Stream 1:** This stream utilizes SQL Data Warehouse/Database (SQL DW/DB) as the data storage and management solution. The team in charge of this stream will leverage the capabilities of Power BI for data visualization and creating interactive dashboards. The combination of SQL DW/DB and Power BI ensures efficient data processing, storage, and analysis, providing stakeholders with valuable insights to support data-driven decision-making.

ii. **Stream 2:** In this stream, the team will employ Azure Databricks with SQL End-point (ADB SQL End-point) as the data processing and analysis platform. Power BI will be used for data visualization and interactive dashboard creation. By leveraging the distributed data processing capabilities of Azure Databricks and combining it with Power BI's visualization

capabilities, this stream enables efficient and scalable data processing, ensuring stakeholders have access to timely and insightful information.

By splitting the case study into these two streams, the project benefits from parallel efforts, maximizing efficiency and expertise in both SQL-based and Databricks-based data processing approaches. This approach allows for a comprehensive exploration of different technologies, resulting in a well-rounded and robust solution for meeting the specified data processing and visualization requirements.

Deliverables:

Create a presentation which has:

Slide 1: BatchName_FirstName_SecondName
Slide 2: Problem statement
Slide 3: Implemented data flow diagram showing various technical components and Layers.
Slide 4-6: Snapshots of developments in each layer (RAW, STAGING(STG), CURATED)
Slide 7: Screenshot of dashboards built on Power BI.
Slide 8: GitHub link where solution is available
Slide 9: System Demo
Slide 10: Q&A
Slide 11: Challenges faced, learnings, suggestions, and feedback.

Rubrics for Case Study Evaluation:

Deliverables / milestones	Remarks	Max Marks
▪ GitHub account creation (5 Marks) ▪ Proposing your own Architecture design and details (15Marks)	Activities	20
▪ Data Management and Storage (10 marks)	Activities	10
▪ Data ingestion and Transformation technique details (20 marks)	Activities	20
▪ Visualization of data, by keeping scope of Business User (10 Marks) ▪ Story telling by visualizing data (10 marks)	Activities	20
▪ Live presentation of Solution on Azure portal (15 marks) ▪ Viva (15 marks)	Activities	30
	Total Marks	**100**