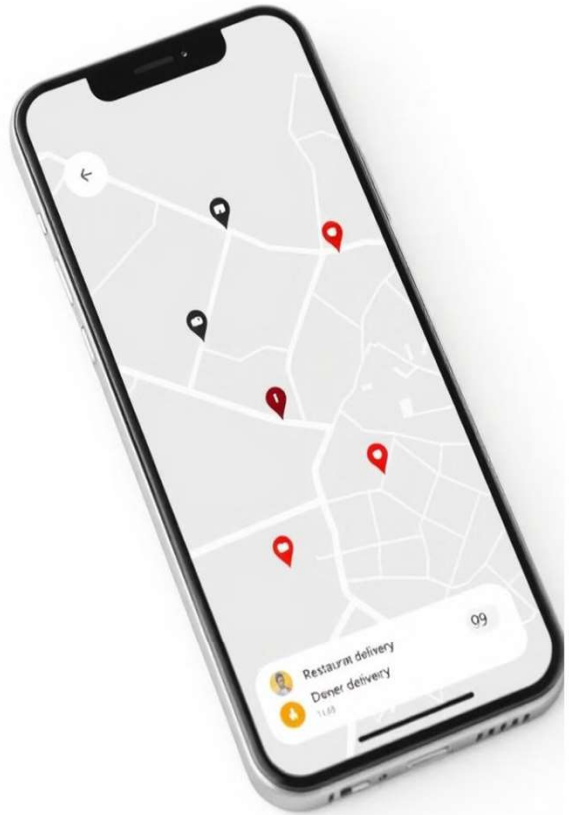
INDIAN INSTITUTE OF TECHNOLOGY JODHPUR

Trimester 3 – Post Graduate Diploma in Data Engineering

Big Data Management

Food Delivery Management System



This presentation provides a comprehensive account of the system's architecture, development process, technical challenges, and the innovations applied. The system leverages Python, Google Cloud Platform (GCP) services such as BigQuery and Google Cloud Storage, and scikit-learn for machine learning tasks. We will delve into the system's architecture, implementation, and key results, highlighting the benefits of using cloud-based solutions for data management and analysis.

By,

Aneerban Chowdhury, G23AI2059

Vanshika Gupta, G23AI2050

Shikha Soni G23AI2075

Project Overview

Problem Statement

The project addresses the challenges of efficiently managing and analyzing large datasets related to food delivery, including customer orders, restaurants, and delivery operations.

Objectives

The system aims to provide a scalable solution for data storage, processing, visualization, and predictive analytics, enabling stakeholders to gain actionable insights and optimize business strategies.

System Architecture

Data Storage

Google Cloud Storage (GCS) and BigQuery are used for storing structured and unstructured data, ensuring efficient management and query capabilities.

Data Processing

Python scripts are employed for data cleaning, transformation, and loading (ETL), ensuring data quality and consistency.

Visualization

Matplotlib and Seaborn libraries are used to create visually appealing charts and graphs, providing clear insights into data trends.

Machine Learning

Scikit-learn is leveraged for predictive analytics, enabling the system to forecast order volumes and identify patterns in historical data.

Frontend

Flask is used to develop a responsive and interactive web interface, allowing users to access key insights and visualizations.

Implementation Steps

1

Environment Setup

The project involved setting up a development environment with Python, essential libraries, and Google Cloud Platform (GCP) services.

2

Data Design

Three main datasets were designed: Restaurants, Customers, and Orders, with appropriate schemas for efficient data storage and querying.

3

Data Capturing and Storage

Data was captured from Kaggle or synthetic sources, stored in CSV files on GCS, and loaded into BigQuery for analysis.

4

Data Analytics

SQL queries were written to analyze data trends, summarize key metrics, and identify insights like high-performing restaurants and customer spending patterns.

5

Machine Learning

A Linear Regression model was developed to predict order volume based on historical data, using scikit-learn for model training and evaluation.

6

Frontend Development

A basic yet functional HTML interface was designed using Flask templates, enabling users to view order summaries and visualizations dynamically.

Key Results

1 Scalability

The system successfully stored and queried large datasets in BigQuery, demonstrating its scalability for handling growing data volumes.

2 Efficiency

Leveraging GCP tools significantly reduced data processing time, improving overall system efficiency and responsiveness.

3 Predictive Insights

The machine learning model achieved acceptable accuracy in predicting order trends, providing valuable insights for operational planning and resource allocation.

4 Accessibility

The frontend interface enabled stakeholders to interact with the system, access key insights, and visualize data in an easily digestible format.

Challenges and Future Scope

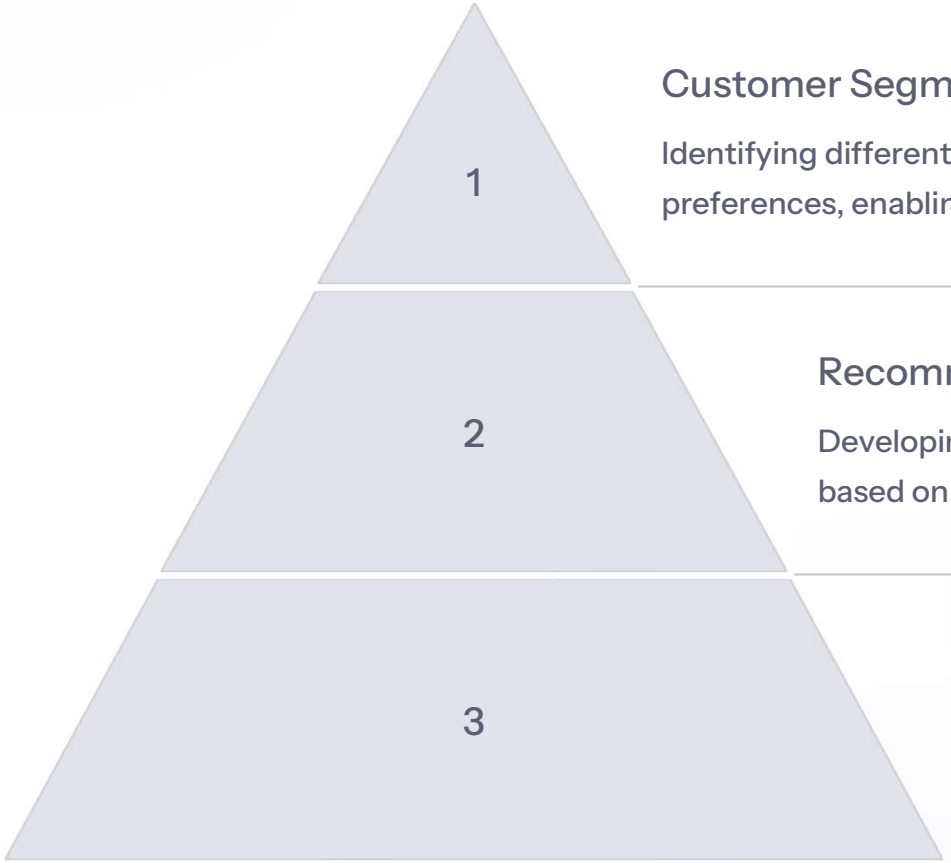
Challenges

The project faced challenges related to data cleaning, ensuring data consistency during synthetic data generation, and balancing clarity and complexity in data visualizations.

Future Scope

Future enhancements include incorporating advanced analytics techniques, real-time data processing, a fully responsive web application, more complex machine learning models, and multi-cloud support.

Advanced Analytics



Customer Segmentation

Identifying different customer groups based on their ordering behavior and preferences, enabling targeted marketing and personalized recommendations.

Recommendation Engines

Developing algorithms to suggest restaurants, dishes, and promotions based on individual customer preferences and past orders.

Predictive Modeling

Building more sophisticated models to forecast demand, identify potential issues, and optimize delivery routes for improved efficiency.

Real-time Data Processing

1

Apache Kafka

Integrating Apache Kafka for real-time data ingestion, enabling the system to process data as it arrives, providing immediate insights and triggering actions.

2

Event-driven Architecture

Adopting an event-driven architecture to respond to real-time events, such as order placement, delivery updates, and customer feedback, for dynamic decision-making.

3

Stream Processing

Implementing stream processing techniques to analyze data in real-time, enabling the system to detect anomalies, identify trends, and provide timely alerts.

Enhanced Frontend

Interactive Dashboard

Developing a comprehensive dashboard with interactive visualizations.

User-friendly Interface

Designing an intuitive and user-friendly interface, simplifying navigation.

Mobile Responsiveness

Ensuring the application is fully responsive across different devices.

Conclusion

The Food Delivery Management System demonstrates the power of cloud-based solutions for managing large-scale data in the food delivery industry. By integrating BigQuery, GCS, and Python, the project showcases a seamless workflow from data ingestion to visualization and predictive analytics. This system provides a robust foundation for future enhancements, enabling stakeholders to gain valuable insights and optimize business strategies.