**INDIAN INSTITUTE OF TECHNOLOGY JODHPUR**

**NH 62 Nagaur Road, Karwar, Jodhpur, Rajasthan**



**Trimester 3 – Post Graduate Diploma in Data Engineering**

**Big Data Management**

**Project Report**

**On**

**Food Delivery Management System**

**Submitted By:**

**Aneerban Chowdhury, G23AI2059**

**Vanshika Gupta, G23AI2050**

**Shikha Soni, G23AI2075**

**Food Delivery Management System**

**Abstract**

The Food Delivery Management System project was undertaken to design and implement an end-to-end solution for managing food delivery data at scale. This report 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 (GCS), and scikit-learn for machine learning tasks. This report also discusses future enhancements, alternative approaches, and the broader implications of this system.

**Introduction**

Food delivery services have seen unprecedented growth, leading to an exponential increase in the volume of data generated daily. This project aims to address the challenges of managing and analysing this data efficiently using modern tools and frameworks. The goal is to design a scalable system for storing, processing, analysing, and visualizing data while incorporating predictive analytics for business insights.

**Problem Statement**

1. **Data Management**: Efficiently handling and processing large datasets encompassing restaurants, customers, and orders. Managing structured and unstructured data in a way that ensures consistency, accuracy, and security. Addressing the challenges of ingestion, storage, and retrieval of high-volume data while maintaining optimal performance.
2. **Analytics**: Deriving actionable insights by analyzing data trends, such as customer purchasing behavior, popular cuisines, and peak ordering times. The goal is to leverage data analytics to guide decision-making processes and improve business strategies for stakeholders.
3. **Prediction**: Building and deploying machine learning models to forecast order volumes accurately. This includes identifying patterns from historical data and predicting future trends to assist in resource allocation and operational planning.
4. **Accessibility**: Designing a user-friendly frontend interface that facilitates seamless access to key insights and visualizations. Ensuring the interface is intuitive and provides clear, actionable information to non-technical users.
5. **Scalability**: Developing a system architecture capable of scaling with increasing data size and user traffic. Incorporating strategies to handle larger datasets and concurrent users without compromising system performance or reliability

**Objectives**

1. **Data Storage**: Utilize Google Cloud Storage (GCS) and BigQuery for efficient storage and management of both structured and unstructured data. Implement schemas for relational datasets, ensuring easy query ability and integration with analytics tools.
2. **Data Visualization**: Create comprehensive and visually appealing reports that represent key metrics and trends. These visualizations should be designed for stakeholders to understand business performance and customer behavior at a glance.
3. **Machine Learning**: Implement predictive models using state-of-the-art machine learning techniques to forecast order trends. Employ linear regression and other algorithms to derive meaningful predictions from historical data.
4. **Frontend**: Develop a responsive and interactive web interface for end-users to interact with the system. This includes features for viewing order summaries, exploring visualized data, and accessing predictive insights in an easily digestible format.

**Methodology**

**System Architecture**

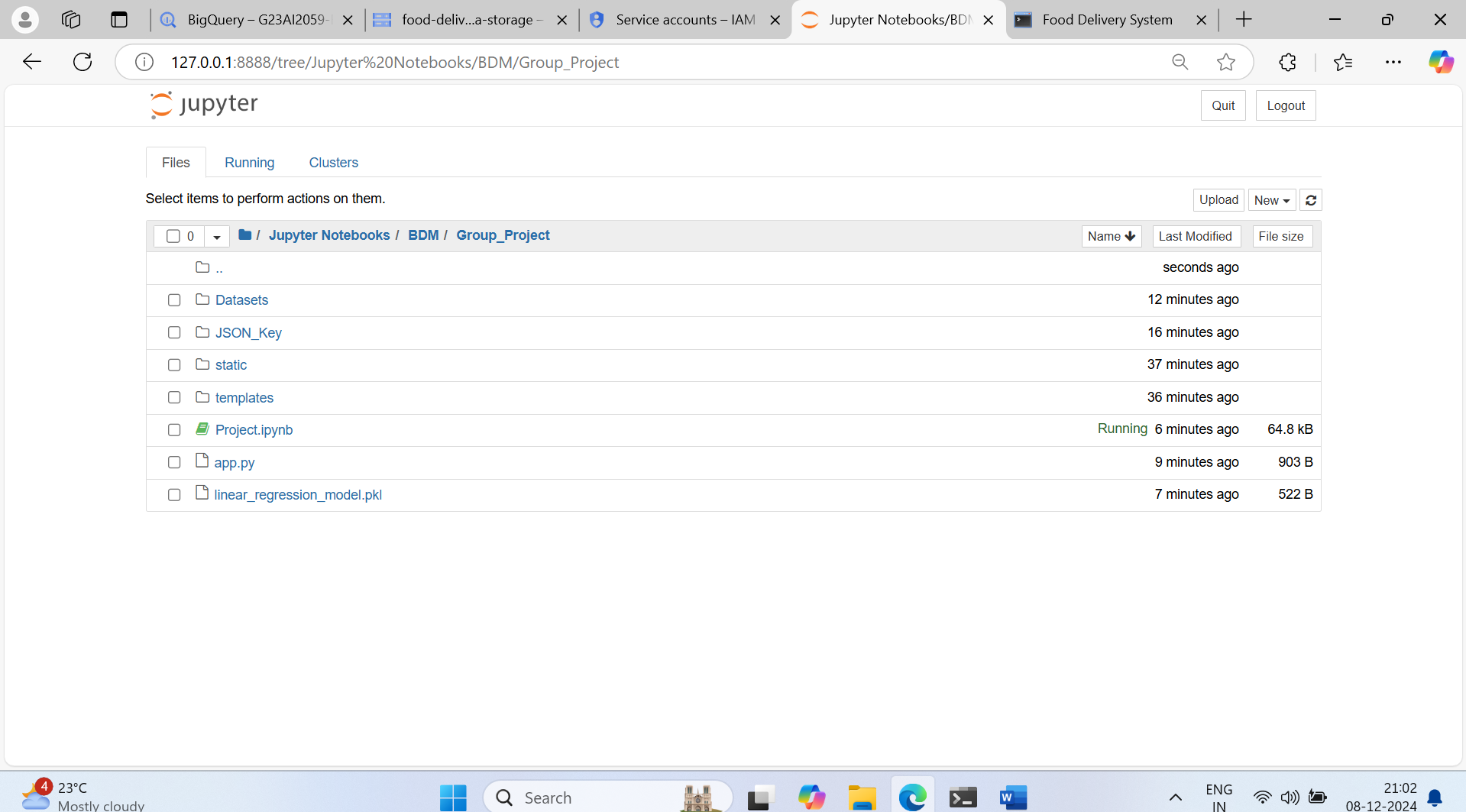
The project is built using a modular architecture that integrates the following components:

1. **Data Storage:** Google Cloud Storage and BigQuery for structured and unstructured data.
2. **Data Processing:** Python scripts for data cleaning, transformation, and loading (ETL).
3. **Visualization:** Matplotlib and Seaborn for charts and graphs.
4. **Machine Learning:** Scikit-learn for predictive analytics.
5. **Frontend:** Flask for web interface development.

**Implementation Steps**

**Step 1: Environment Setup**

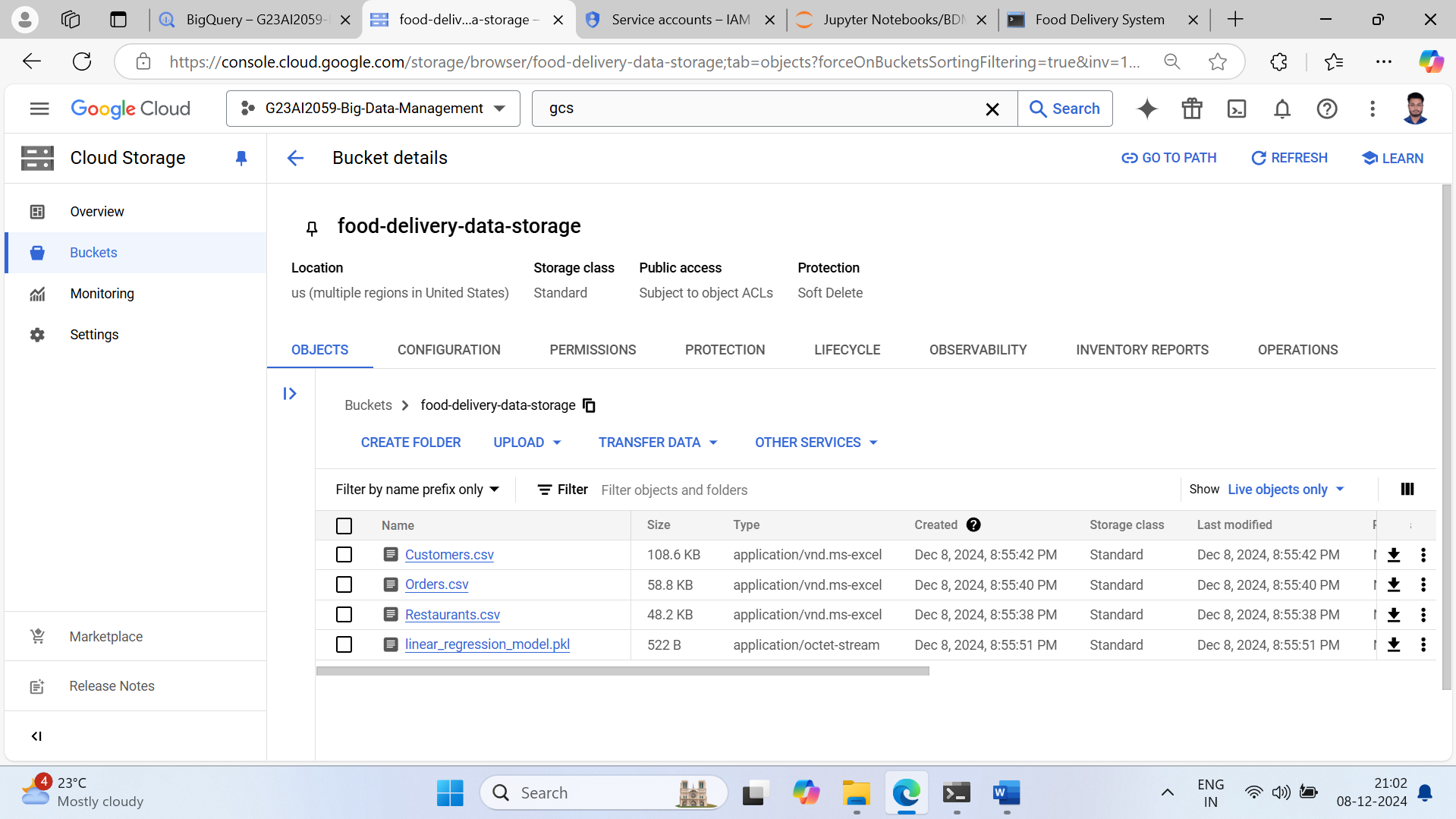
* Installed Python and essential libraries like google-cloud-bigquery, google-cloud-storage, and scikit-learn.
* Set up GCP with appropriate permissions and billing.
* Configured Jupyter Notebook for development and debugging

****

**Step 2: Data Design**

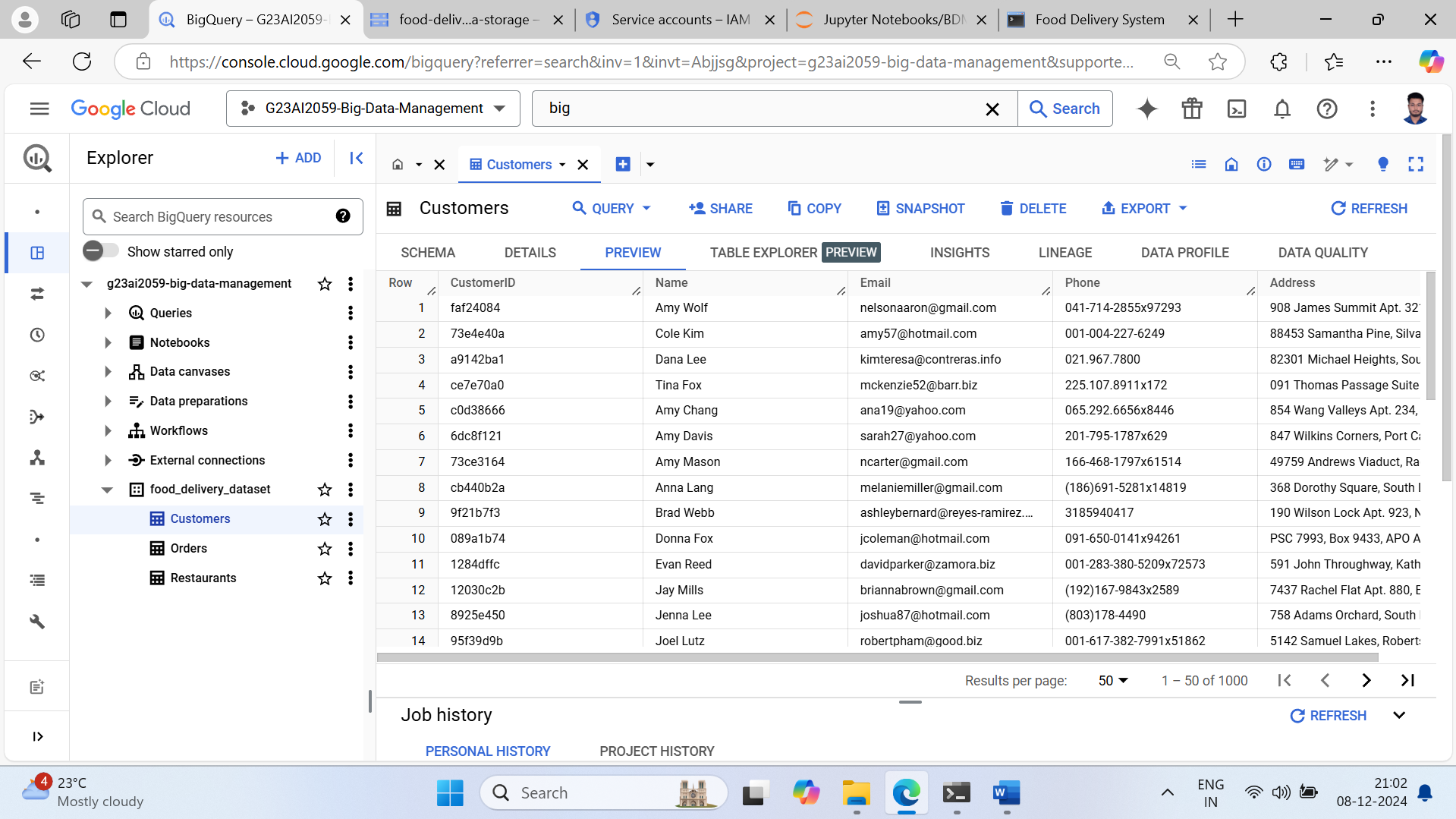
Designed three main datasets:

* **Restaurants:** Contains details like RestaurantID, Name, Location, CuisineType, and Rating.
* **Customers:** Includes CustomerID, Name, Email, Phone, and Address.
* **Orders:** Records OrderID, CustomerID, RestaurantID, OrderDate, OrderStatus, and OrderAmount.

****

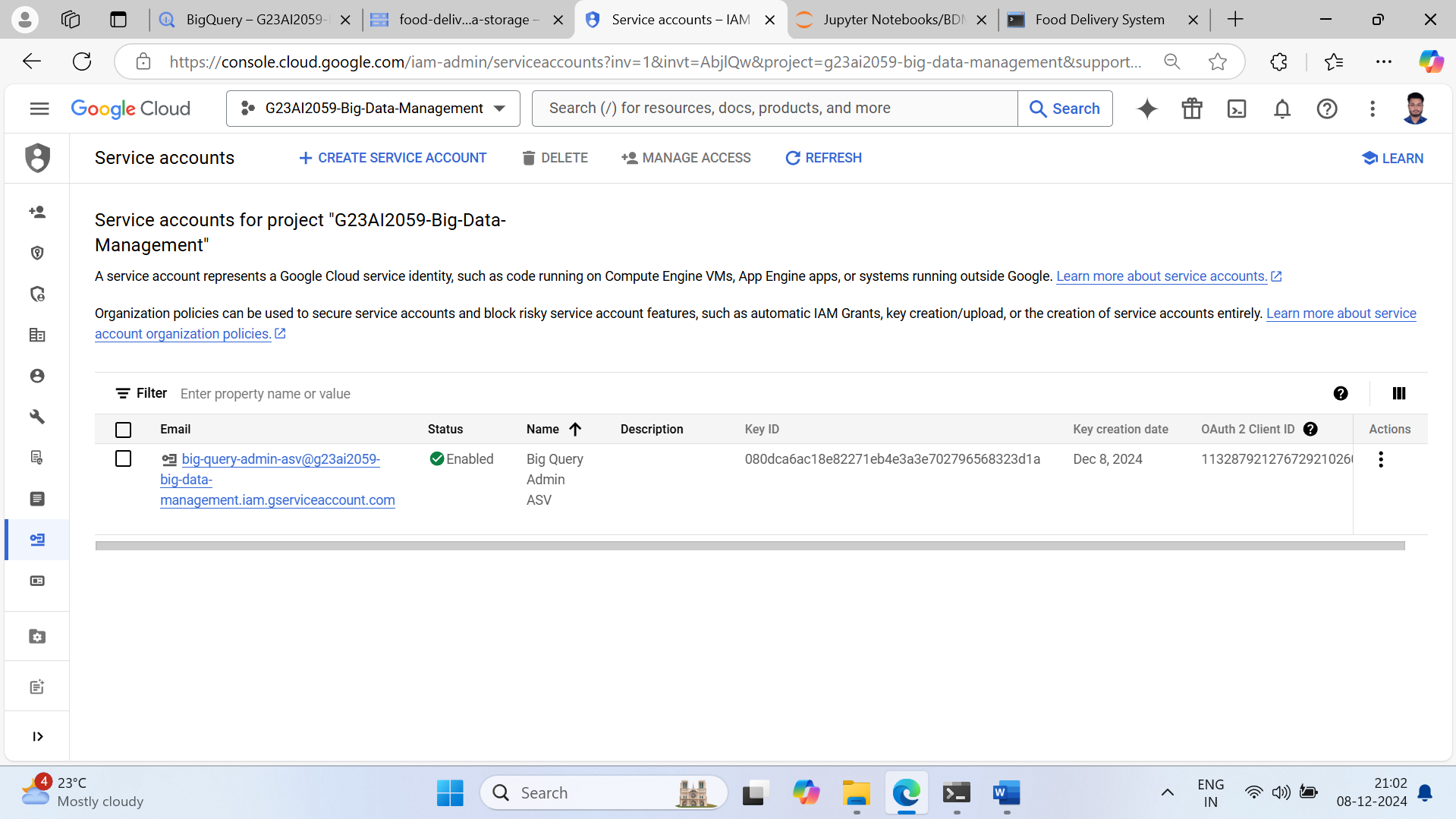
**Step 3: Data Capturing and Storage**

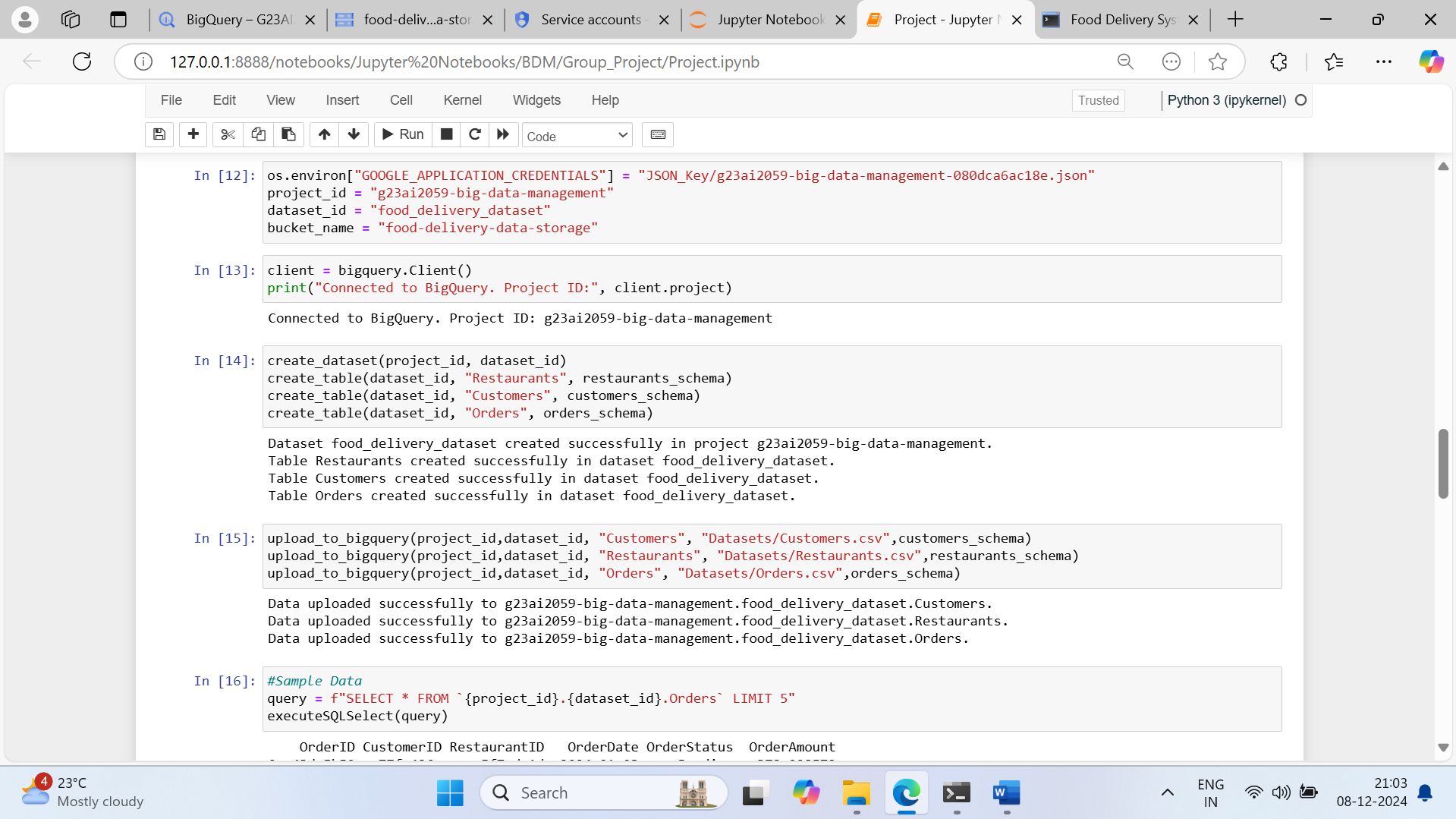
* Capture data from Kaggle or synthetic data sources 1,000 rows for each table.
* Stored CSV files in GCS and loaded them into BigQuery for querying.

****

**Step 4: Data Analytics**

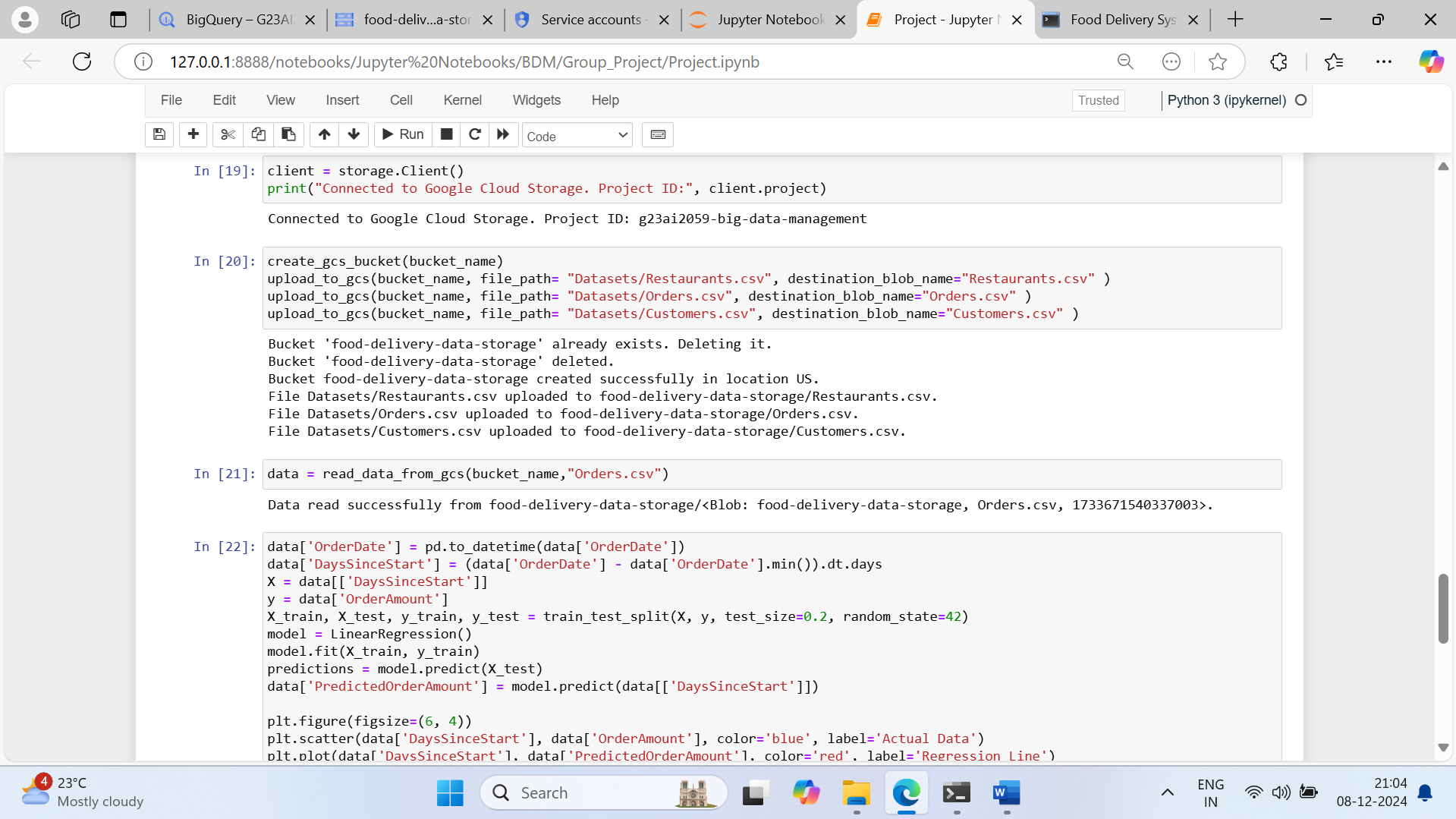
* Wrote SQL queries to analyze trends and summarize data.
* Explored insights like high-performing restaurants, customer spending patterns, and peak order times.

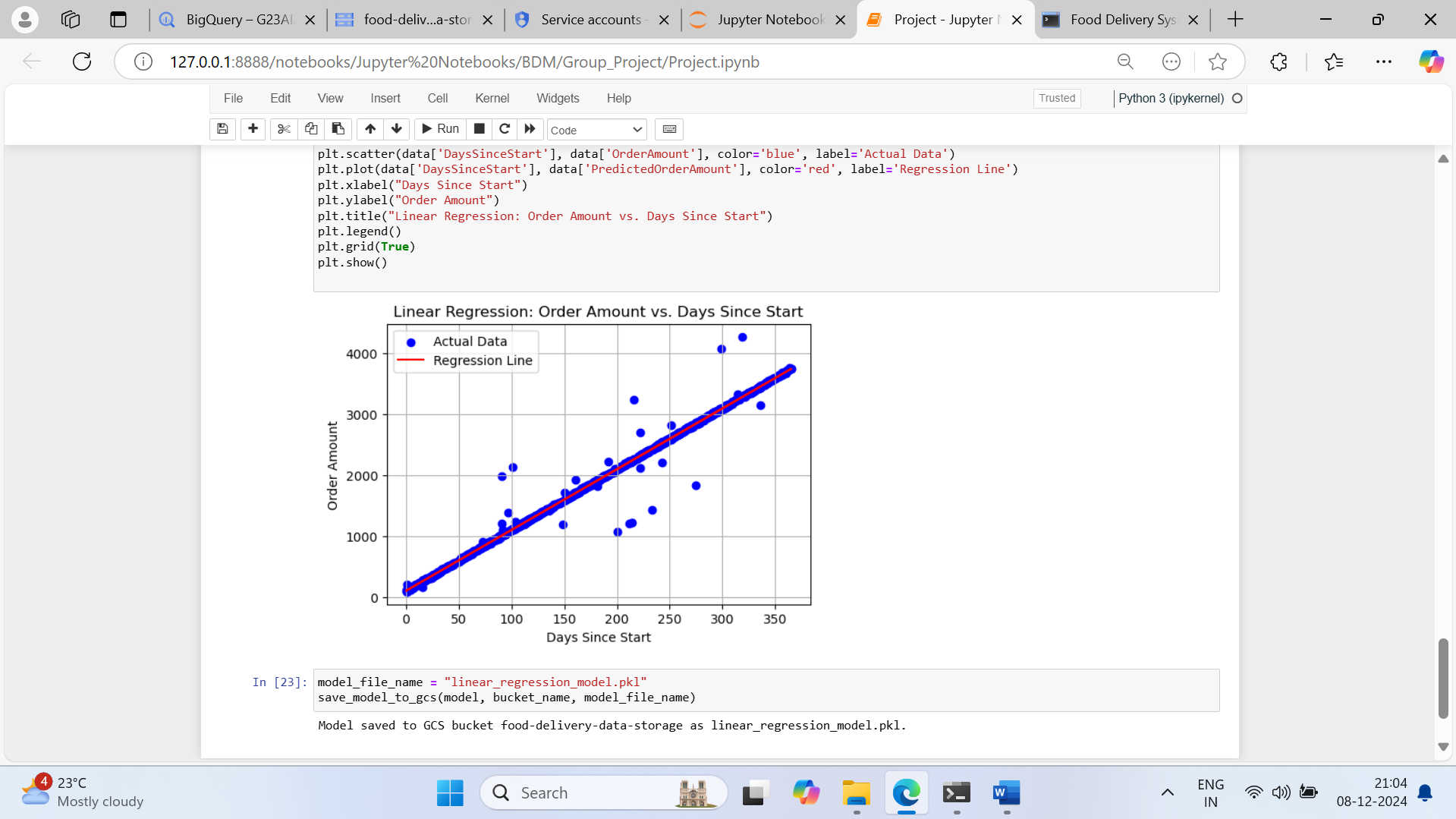
****



**Step 5: Machine Learning**

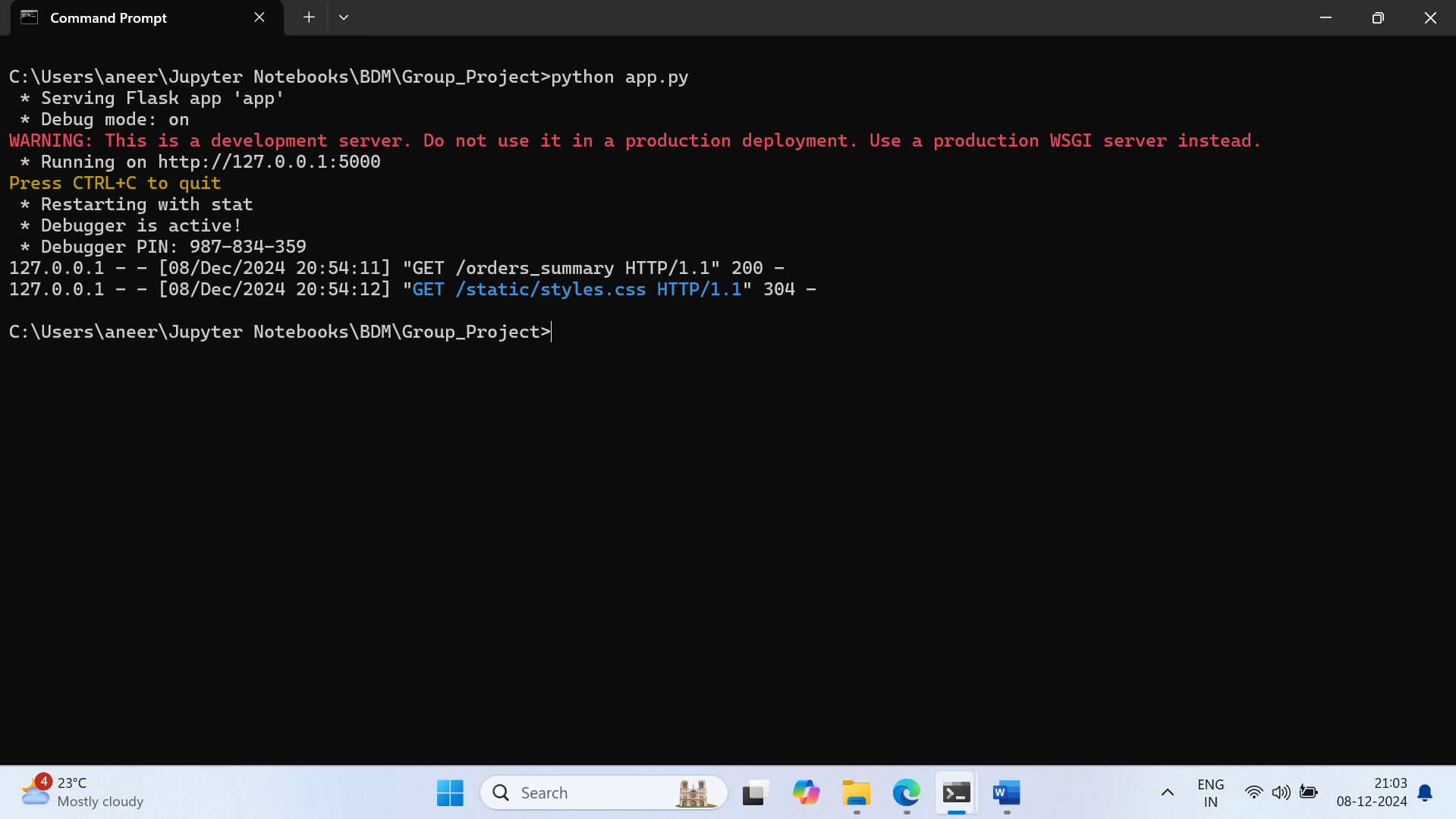
* Developed a Linear Regression model to predict order volume based on historical data.
* Evaluate the model using scikit-learn.

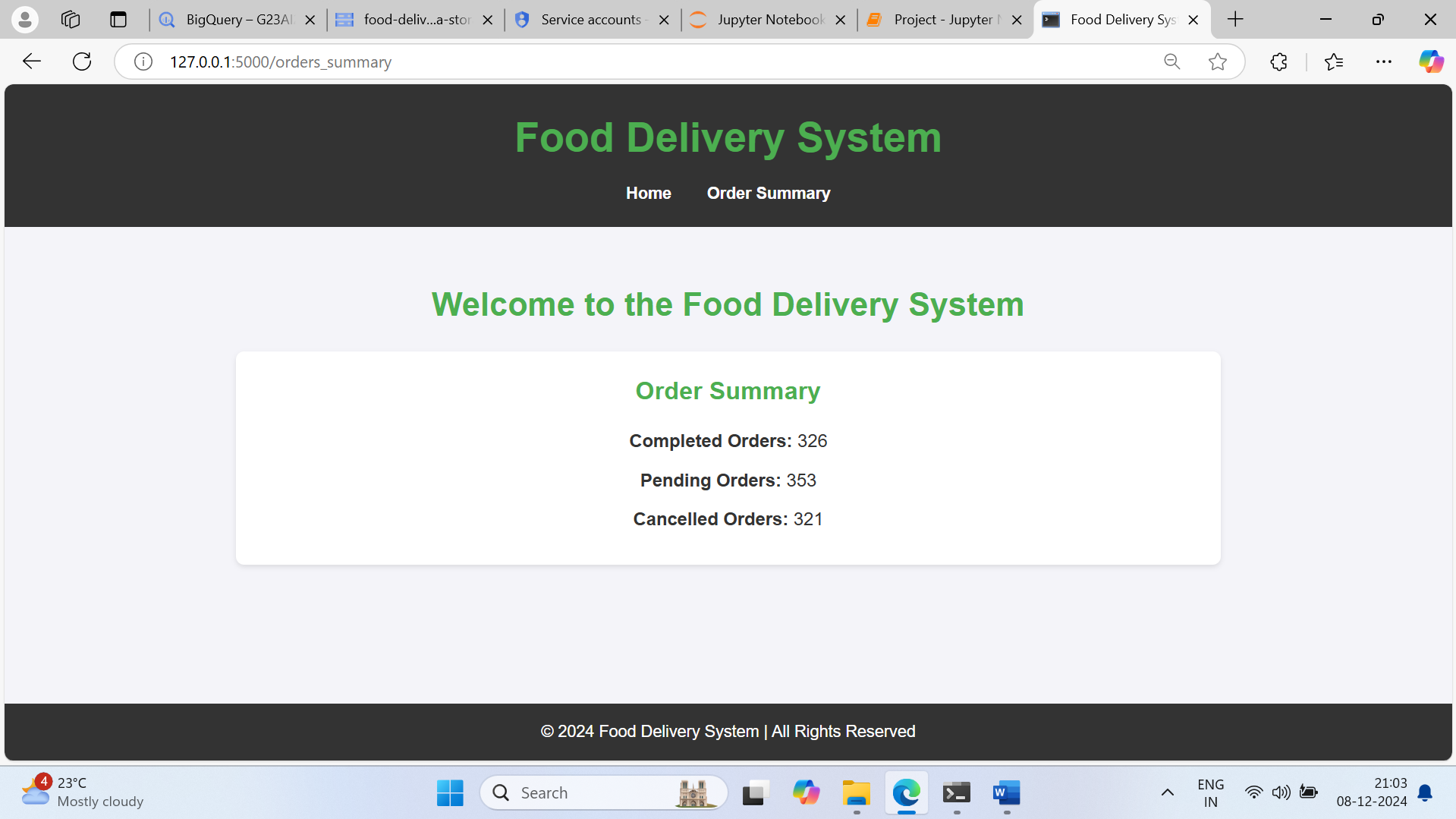


****

**Step 6: Frontend**

* Designed a basic yet functional HTML interface integrated with Flask templates.
* Displayed order summaries and visualizations dynamically.

****

****

**Results**

**Achievements**

1. **Scalability:** Successfully stored and queried large datasets in BigQuery.
2. **Efficiency:** Reduced data processing time using GCP tools.
3. **Predictive Insights:** Achieved acceptable accuracy in predicting order trends.
4. **Accessibility:** Enabled stakeholders to interact with the system through frontend.

**Challenges**

1. **Data Cleaning:** Ensuring data consistency during synthetic data generation.
2. **Visualization:** Balancing clarity and complexity in visual outputs.

**Future Scope**

1. **Advanced Analytics:** Incorporate customer segmentation and recommendation engines.
2. **Real-time Processing:** Use tools like Apache Kafka for real-time data ingestion.
3. **Enhanced Frontend:** Develop a fully responsive web application.
4. **Scalable ML Models:** Integrate more complex models for better predictions.
5. **Multi-cloud Support:** Extend functionality to other cloud platforms.

**Conclusion**

The Food Delivery Management System demonstrates the potential of cloud-based solutions for managing large-scale data. By integrating BigQuery, GCS, and Python, the project showcases a seamless workflow from data ingestion to visualization. This system not only meets the current requirements but also provides a robust foundation for future enhancements.

Project Link: [**https://github.com/aneerban10/BDM-Group-Project-Aneerban-Shikha-Vanshika**](https://github.com/aneerban10/BDM-Group-Project-Aneerban-Shikha-Vanshika)

**References**

1. Google Cloud Platform Documentation
2. Scikit-learn Reference
3. Flask Documentation