KLE Society's

KLE Technological University



**An Industry Project Report**

**On**

**Online Food Delivery System**

*Submitted in partial fulfillment of the requirement for the degree of*

**Bachelor of Engineering in**

**Computer Science and Engineering**

**Submitted By**

**Manish**

**01FE21BCS149**

**Under the guidance of**

**Mr. Vijay Biradar**

COMPUTER SCIENCE &ENGINEERING,

HUBBLLI–580 031 (India).

Academic year 2024-25



**B. V. Bhoomaraddi College Campus, Vidyanagar, Hubballi - 580031.**

**Karnataka (India)**

**COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that Industry Project entitled “**Online Food Delivery System**” is a Bonafide work carried out by the student Mr. Manish bearing USN 01FE21BCS149 in partial fulfillment of the completion of 8th semester B.E. course during the year 2024 – 25 at **Cognizant**. The Industry Project report has been approved as it satisfies the academic requirement with respect to the project work prescribed for the above said course.

**Name of the Guide HoD**

Mr. Vijay Biradar

**Name of the examiners Signature with date**

1 -------------------------- ------------------ 1 -------------------------- ---------------

2 -------------------------- ----------------- 2 -------------------------- ---------------

**CERTIFICATE**

**A white and blue email

AI-generated content may be incorrect.**

**DECLARATION**

I hereby declare that the Industry Project Report entitled “**Online Food Delivery System**” is an authentic record of my own work as requirements of Industry, during the period from 21-04-2025 to 15-05-2025 for the award of the degree of B.E. Under the guidance of Mr. Vijay Biradar.

**A close up of a sign

AI-generated content may be incorrect. Manish**

**Date: 01-06-2025**

**01FE21BCS149**

**ACKNOWLEDGEMENT**

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of a number of individuals whose professional guidance and encouragement helped me in the successful completion of this report work.

I also take this opportunity to thank Dr. Vijaylakshmi M, Professor and Head, Department of Computer Science and Engineering for having provided us academic environment which nurtured our practical skills contributing to the success of our project.

I sincerely thank our guide Mr. Vijay Biradar for his guidance and wholehearted co-operation during the course of completion.

I sincerely thank Vighnesh , Cognizant for her support, inspiration and wholehearted co-operation during the course of completion.

My gratitude will not be complete without thanking our beloved parents, our seniors and our friends who have been a constant source of aspirations.

**Manish**

**ABSTRACT**

In the era of digital transformation, organizations are increasingly reliant on data-driven strategies to understand customer behavior, personalize experiences, and optimize business operations. This project focuses on the development of a **Customer Data Platform (CDP)** using **Google Cloud Platform (GCP)** to address the growing need for a unified, secure, and scalable data infrastructure.

The CDP is designed to ingest real-time and batch data from diverse sources such as CRM systems, web analytics platforms, and transactional databases. Leveraging GCP-native services like **Cloud Pub/Sub**, **Dataflow**, **BigQuery**, **Cloud KMS**, **IAM**, and **Looker Studio**, the platform ensures seamless data ingestion, transformation, secure storage, and insightful visualization. Sensitive data fields are encrypted using **Cloud Key Management Service (KMS)** to maintain compliance with data privacy regulations such as GDPR and CCPA.

The system architecture supports both streaming and batch processing, enabling near real-time analytics and historical data analysis. Key Performance Indicators (KPIs) are computed using BigQuery and visualized through interactive dashboards in Looker Studio, providing stakeholders with actionable insights into customer engagement, transaction trends, and operational performance.

Additionally, the platform incorporates robust monitoring and alerting mechanisms using **Cloud Monitoring and Logging**, ensuring high availability, fault tolerance, and operational transparency. This project not only demonstrates the technical feasibility of building a cloud-native CDP but also highlights its strategic value in enabling data-driven decision-making, enhancing customer experience, and improving business agility.

i

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Table of Contents** | **Page No.** |
| **1.** | **Introduction** | **1** |
| 1.1 | Literature Survey | **1** |
| 1.2 | Motivation | **2** |
| 1.3 | Objectives | **3** |
| 1.4 | Problem Definition | **3** |
| **2.** | **Requirement Analysis** | **4** |
| 2.1 | System Model | **4** |
| 2.2 | Functional Requirements | **4** |
| 2.3 | Non-Functional Requirements | **5** |
| 2.4 | Database requirements | **5** |
| **3.** | **System Design** | **6** |
| 3.1 | Architecture Design | **6** |
| 3.2 | Data Flow Diagram | **7** |
| **4.** | **Implementation** | **8** |
| 5. | Results and Discussions | **23** |
| 6. | Conclusion and Future Scope | **33** |
| 7. | References | **34** |

ii

**List of Figures**

1. **Figure 1** – High Level Architecture
2. **Figure 2** – Data Flow Diagram
3. **Figure 3** – Buckets
4. **Figure 4** – Data in respective buckets
5. **Figure 5** – BigQuery Datasets
6. **Figure 6** – Tables in CRM
7. **Figure 7** – Tables in Web
8. **Figure 8** – Tables in Transaction
9. **Figure 9** – Code for Publishing Data to Pub/Sub Topic
10. **Figure 10** – Employee Streaming Dataflow Job
11. **Figure 11** – Data Streamed into BigQuery Table
12. **Figure 12** – Customer.js
13. **Figure 13** – Customer.json
14. **Figure 14** – Customer Streaming Dataflow Job
15. **Figure 15** – Customer Table Schema
16. **Figure 16** – Transformed Customer Data
17. **Figure 17** – Click\_Events.js
18. **Figure 18** – Click\_Events.json
19. **Figure 19** – Click Events Streaming Dataflow Job
20. **Figure 20** – Click Events Table Schema
21. **Figure 21** – Transformed Click Events Data
22. **Figure 22** – Key Generation in KMS
23. **Figure 23** – Python Code for Encryption
24. **Figure 24** – Bucket Containing Encrypted Data
25. **Figure 25** – Encrypted Transactional Data in BigQuery
26. **Figure 26** – Encrypted Orders Data
27. **Figure 27** – Encrypted Orders Data
28. **Figure 28** – Encrypted Order Items Data
29. **Figure 29** – Encrypted Payments Data
30. **Figure 30** – Permissions Provided for Service Account
31. **Figure 31** – Storage Admin Role
32. **Figure 32** – Bucket Accessed by User
33. **Figure 33** – KPI 1
34. **Figure 34** – KPI 2
35. **Figure 35** – KPI 3
36. **Figure 36** – KPI 4
37. **Figure 37** – KPI 5
38. **Figure 38** – KPI 6
39. **Figure 39** – KPI 7
40. **Figure 40** – KPI 8
41. **Figure 41** – Customer Overview Dashboard
42. **Figure 42** – Web Analytics Dashboard
43. **Figure 43** – Transaction Insights Dashboard
44. **Figure 44** – Monitoring Dashboard
45. **Figure 45** – Cloud Storage Monitoring
46. **Figure 46** – BigQuery Error Alerting
47. **Figure 47** – Logging Dashboard

iii

Chapter 1:

# Introduction

The Customer Data Platform (CDP) on Google Cloud Platform (GCP) is a cloud-native solution designed to centralize and streamline the ingestion, processing, and analysis of customer and web analytics data. This platform enables real-time data handling, secure storage, and insightful visualization of business metrics. By leveraging GCP-native services such as Pub/Sub, Dataflow, BigQuery, Cloud KMS, IAM, and Looker Studio, the CDP ensures scalability, security, and operational efficiency. The system is tailored to support data analysts, engineers, and business managers in making informed decisions based on real-time insights.

Chapter 1.1:

# Literature Survey

**1. The Emerging Challenges of Customer Relationship Management with Customer Data Platforms (CDP): Constraints and Perspectives**

* Authors: Chen Jian, Dan Tshiswaka Dan
* Year: 2024
* Key Points:
  + Emphasizes the shift from third-party to first-party data strategies due to the phasing out of third-party cookies.
  + Highlights the use of real-time data activation, cross-channel data aggregation, and privacy compliance tools like GDPR and CCPA.
  + CDPs are positioned as essential for delivering personalized customer experiences and building trust through ethical data practices.
  + Discusses challenges such as integration complexity, regulatory compliance, and consumer demand for transparency and control.
  + Technologies include data unification engines, identity resolution, and consent management frameworks

**2. Customer Data Platform Industry Update – January 2024**

* Author: CDP Institute
* Year: 2024
* Key Points:
  + Reports a trend toward home-built CDP architectures and modular, cloud-native platforms.
  + Notes increased use of API-based integrations, microservices, and composable CDP components.
  + Despite slowed growth in 2023, the market is projected to reach $2.5 billion in 2024.
  + Emphasizes the shift toward cost-efficient, scalable solutions and vendor flexibility.
  + Highlights the growing importance of data governance, real-time analytics, and interoperability with existing enterprise systems

**3. Critical Capabilities for Customer Data Platforms**

* Author: Gartner Research
* Year: 2023
* Key Points:
  + Evaluates CDPs across 13 critical capabilities, including data ingestion, identity resolution, segmentation, and activation.
  + Stresses the need for real-time personalization, omnichannel orchestration, and enterprise system integration (e.g., CRM, ERP).
  + Identifies data quality management, AI-driven insights, and customer journey mapping as emerging differentiators.
  + Technologies assessed include streaming data pipelines, machine learning models, and cloud-native infrastructure

**4. Customer Data Platforms: A Foundation for Customer-Centric Marketing**

* Author: David Raab (CDP Institute)
* Year: 2022
* Key Points:
  + Defines CDPs as platforms that unify customer data using ETL pipelines, cloud data warehouses (e.g., BigQuery, Snowflake), and customer journey orchestration tools.
  + Categorizes CDPs into four types: data-focused, analytics-focused, campaign-focused, and delivery-focused.
  + Highlights use cases such as churn prediction, campaign optimization, real-time personalization, and cross-channel engagement.
  + Emphasizes the importance of data governance, scalability, and integration with marketing automation platforms.

Chapter 1.2:

# Motivation

In the current digital era, businesses are inundated with vast amounts of customer data generated from various sources such as websites, mobile applications, CRM systems, and transactional platforms. To remain competitive, organizations must not only collect this data but also process and analyze it in real time to gain meaningful insights. Traditional data systems often lack the flexibility and scalability required to handle such dynamic and diverse data streams. This creates a pressing need for a modern, cloud-native solution that can unify customer data, ensure data privacy, and support real-time analytics.

The motivation behind this project is to develop a robust and scalable **Customer Data Platform (CDP)** using **Google Cloud Platform (GCP)** that addresses these challenges. The platform is designed to ingest and process large volumes of structured and semi-structured data from multiple sources, providing a unified view of customer interactions. By integrating services like Pub/Sub, Dataflow, BigQuery, and Looker Studio, the CDP enables secure data handling, real-time KPI tracking, and interactive dashboard visualizations. This empowers businesses to make data-driven decisions, enhance customer engagement, and improve operational efficiency.

Chapter 1.3:

# Objectives

The primary objectives of this project are:

* To design and implement a cloud-native Customer Data Platform using GCP services.
* To ingest real-time and batch data from CRM, web analytics, and transactional systems.
* To ensure secure storage and processing of sensitive data using encryption and access control.
* To compute and visualize key business metrics through interactive dashboards.
* To establish monitoring and alerting mechanisms for system reliability and performance.

Chapter 1.4:

# Problem Definition

Organizations often struggle with fragmented customer data spread across multiple systems, leading to inefficiencies in analysis and decision-making. The lack of a centralized platform for real-time data processing and visualization hampers the ability to respond quickly to customer needs. This project addresses the problem by developing a Customer Data Platform on GCP that unifies data ingestion, processing, security, and visualization. The solution aims to provide a single source of truth for customer data, enabling timely insights and strategic actions.

Chapter 2:

# Requirement Analysis

The requirement analysis outlines the essential components, behaviors, and constraints of the system to ensure it meets the intended objectives effectively.

Chapter 2.1:

# System Model

The system is designed as a cloud-native architecture built entirely on **Google Cloud Platform (GCP)**. It follows a modular pipeline approach where data is ingested in real time using **Cloud Pub/Sub**, processed using **Dataflow**, and stored securely in **BigQuery**. Sensitive data fields are encrypted using **Cloud KMS**, and access is managed through **IAM**. The processed data is visualized using **Looker Studio** dashboards. The system supports both batch and streaming data pipelines and is designed for scalability, security, and real-time analytics.

Chapter 2.2:

# Functional Requirements

* Real-time ingestion of CRM and web analytics data using **Cloud Pub/Sub**.
* Data transformation and cleaning using **Cloud Dataflow**.
* Secure storage of structured data in **BigQuery**.
* Encryption of sensitive fields using **Cloud KMS**.
* Implementation of 8 business KPIs using SQL queries in BigQuery.
* Creation of 3 interactive dashboards using **Looker Studio**.
* Email alerting on BigQuery job failures using **Cloud Monitoring**.
* Access control and data protection using **IAM** and **VPC Service Controls**.

Chapter 2.3:

# Non-Functional Requirements

* **Scalability**: The system must handle increasing volumes of data without performance degradation.
* **Security**: All sensitive data must be encrypted and access controlled.
* **Availability**: The system should ensure high availability and fault tolerance.
* **Performance**: Real-time data ingestion and near real-time dashboard updates.
* **Maintainability**: Modular design to allow easy updates and debugging.

Chapter 2.4:

# Database Requirements

The Customer Data Platform (CDP) utilizes **Google BigQuery** as the central data warehouse for storing and analyzing large volumes of structured and semi-structured data. The database design is organized into three primary datasets, each corresponding to a major data source:

**1. CRM Dataset**

* Contains 9 tables:
  + customers, addresses, demographics, preferences, loyalty\_program, engagement, customer\_feedback, customer\_segments, data\_login\_activity
* Data format: CSV
* Purpose: Stores customer profile and behavioral data.

**2. Web Analytics Dataset**

* Contains 5 tables:
  + click\_events, conversion\_events, device\_information, page\_views, session\_data
* Data format: JSON
* Purpose: Captures user interaction data from web platforms.

**3. Transactional Dataset**

* Contains 15 tables:
  + orders, payments, invoices, shipments, returns, refunds, shipment\_tracking, order\_returns, payment\_transaction, order\_items, inventory, product\_catalog, promotions, supplier\_data, inventory\_logs
* Data format: CSV
* Purpose: Maintains records of financial transactions, logistics, and product data.

Chapter 3:

**System Design**

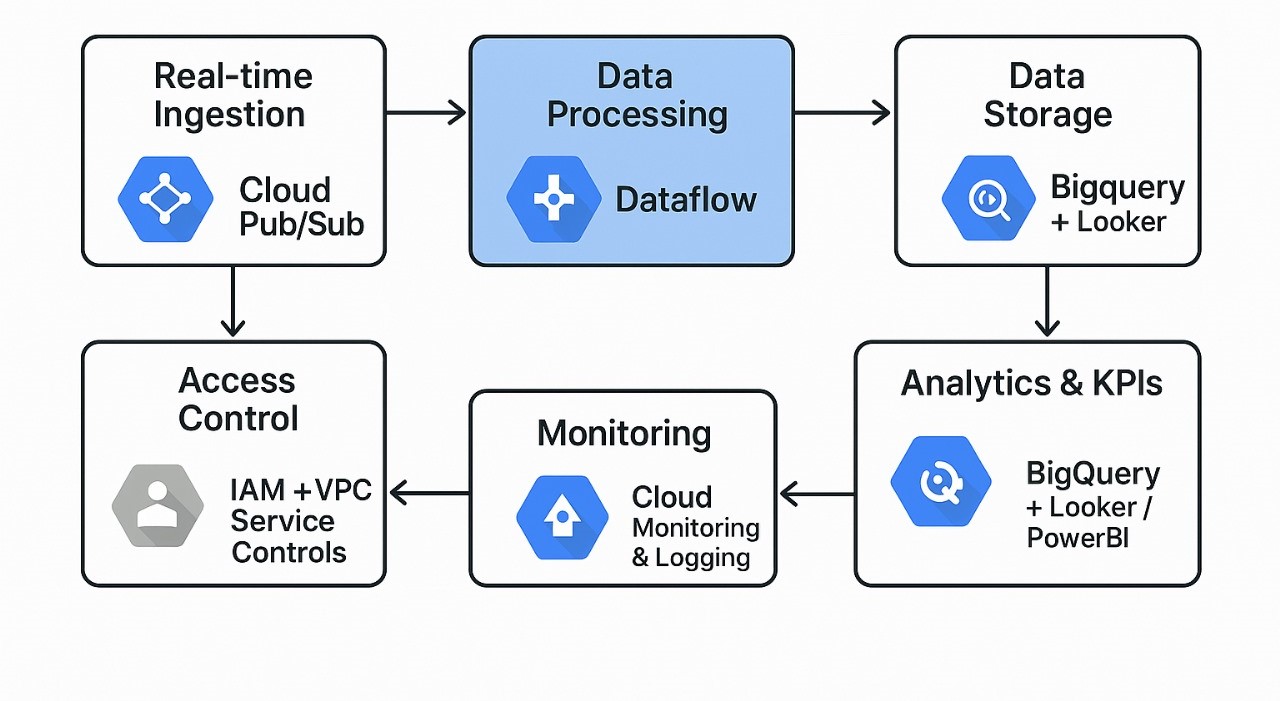
Chapter 3.1:

**Architecture Design**

# A diagram of a cloud monitoring system AI-generated content may be incorrect.(1) High Level Architecture

Chapter 3.2:

**Data Flow Diagram**



# (2) Data Flow Diagram

Chapter 4:

**Implementation**

# Data Sources

* **CRM Data** (CSV format) – 9 tables -> customers, addresses, demographics, preferences, loyalty program, engagement, customer feedback, customer segments, data login activity.
* **Web Analytics Data** (JSON format) - 5 tables - > click events, conversion events, device information, page views, session data.
* **Transactional Data** (CSV format) – 15 tables -> orders, payments, invoices, shipments, returns, refunds, shipment tracking, order returns, payment transaction, order items, inventory, product catalog, promotions, supplier data, inventory logs.

**Note: All data are generated using faker library of python.**

Data Ingestion and Storage

Raw data files were initially stored in Google Drive. These files were programmatically transferred to Google Cloud Storage (GCS) Buckets namely crm\_sample, web\_analy, suhas\_transaction using a Python script executed in Google Colab. This automated approach ensured secure and efficient migration of data from Drive to the cloud.

Once the data was available in GCS, it was imported into BigQuery by creating three separate datasets corresponding to the source systems:

**CRM**

**Web**

**Transaction**

Each dataset was structured with appropriate tables reflecting the schema of the original source data. This served as the raw data layer for the analytics workflow.

A screenshot of a computer

AI-generated content may be incorrect.

(3) Buckets

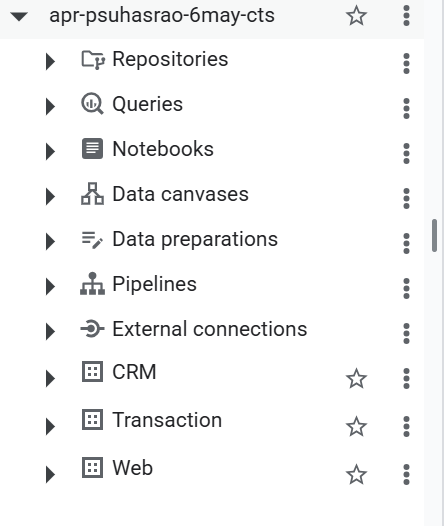
A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

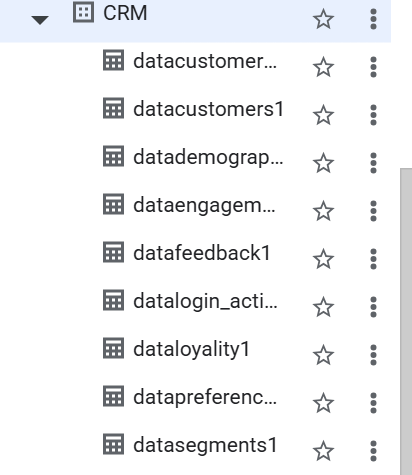
AI-generated content may be incorrect.

# A screenshot of a computer AI-generated content may be incorrect.

(4) Data in respective buckets



(5) BigQuery Datasets



(6) Tables in CRM (7) Tables in Web

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

(8) Tables in Transaction

Real-time Streaming Simulation Using Pub/Sub and Dataflow

To simulate a real-time data streaming pipeline, I utilized Google Cloud Pub/Sub and Dataflow. However, due to the high cost and time involved in streaming the full dataset of 20 lakh records, a scaled-down version was implemented using a sample CSV file (employee.csv) for demonstration purposes.

**Pub/Sub Topic Creation and Data Publishing**

* A Pub/Sub topic named emp\_ex was created.
* A Python script was developed to read the employee.csv file and publish records one by one to the emp\_ex topic.
* A delay of 10 seconds was introduced between each record to simulate real-time ingestion behavior.

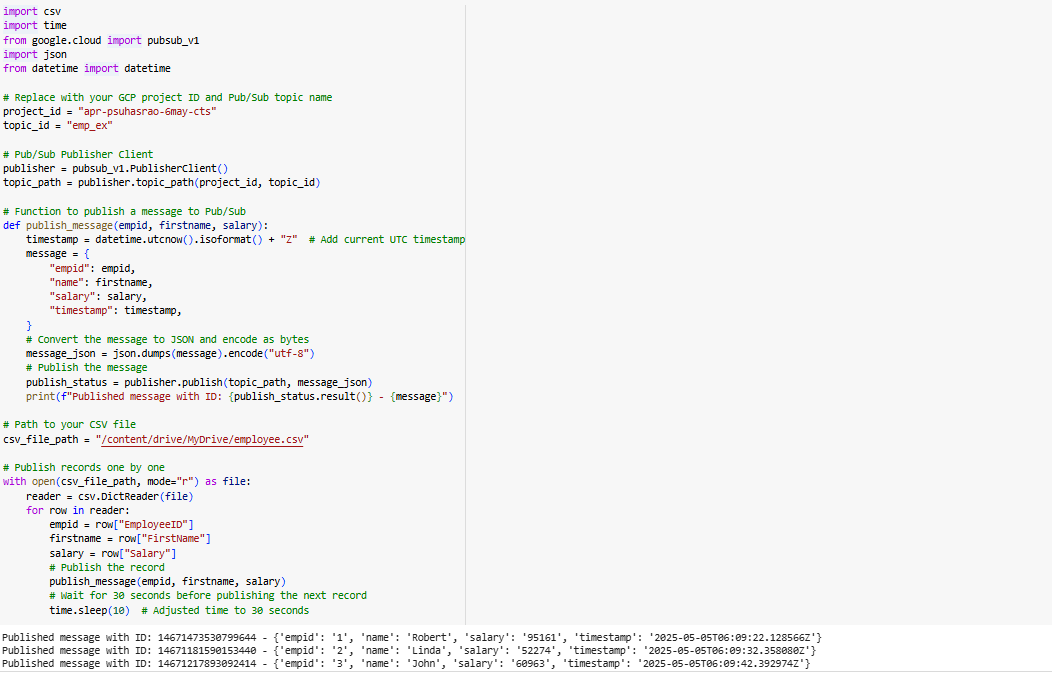
**Streaming Dataflow Job**

* An **empty table** was created in BigQuery with the schema corresponding to the structure of the employee data.
* A **Dataflow job** was launched using the **"Pub/Sub to** **BigQuery" template**.

Input: Pub/Sub topic emp\_ex

Output: BigQuery table

* As the script ran, data published to Pub/Sub **was ingested in near real-time** by Dataflow and populated into the BigQuery table.



(9) Code For Publishing Data to Pub/Sub Topic

A screenshot of a computer

AI-generated content may be incorrect.

(10) Employee Streaming Dataflow Job

A screenshot of a computer

AI-generated content may be incorrect.

(11) Data Streamed into BigQuery Table

Data Processing with Dataflow (Batch Mode)

Although the intended architecture involved real-time streaming using Pub/Sub and Dataflow, the available data was static. Hence, a batch processing approach was adopted to simulate the ingestion pipeline.

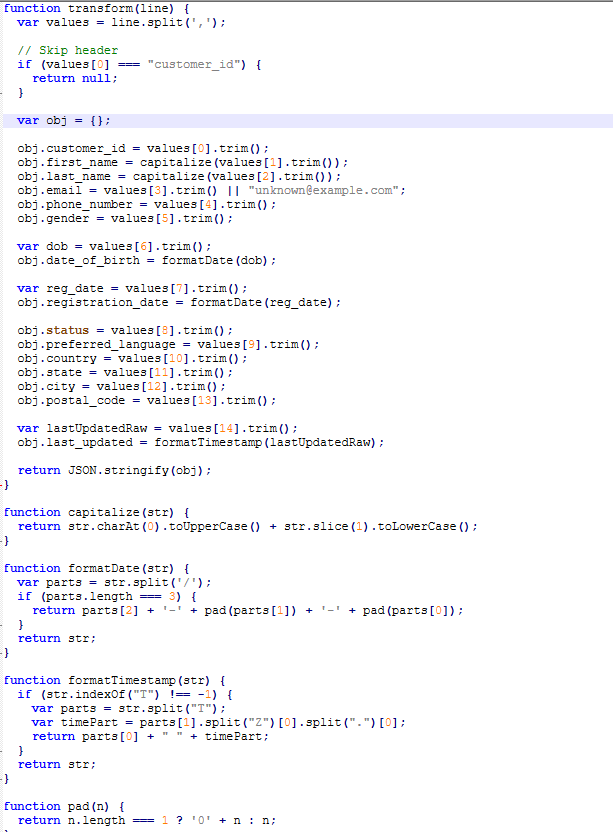
Each source (e.g., Customer, Click Events) included:

* A **JavaScript** **(.js) transformation file** defining how each record should be parsed and cleaned.
* A **JSON schema file** specifying the structure for the BigQuery table.

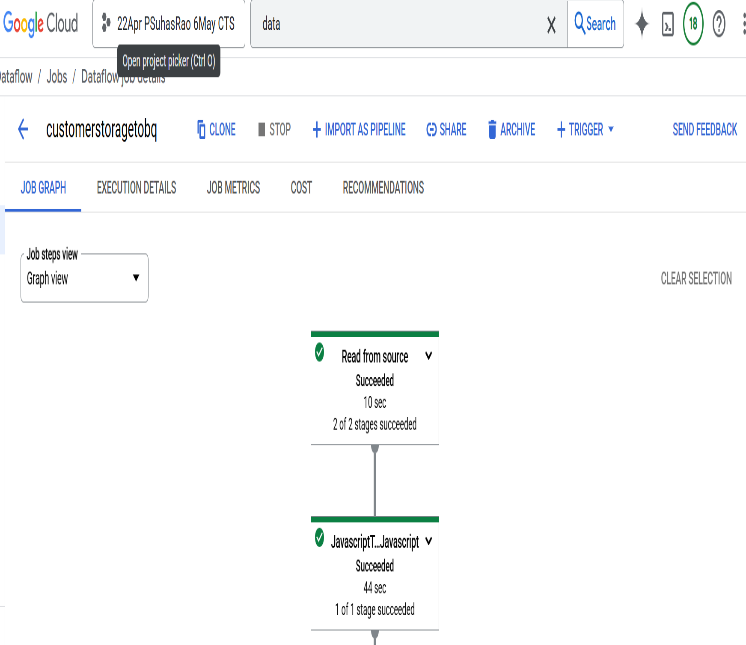
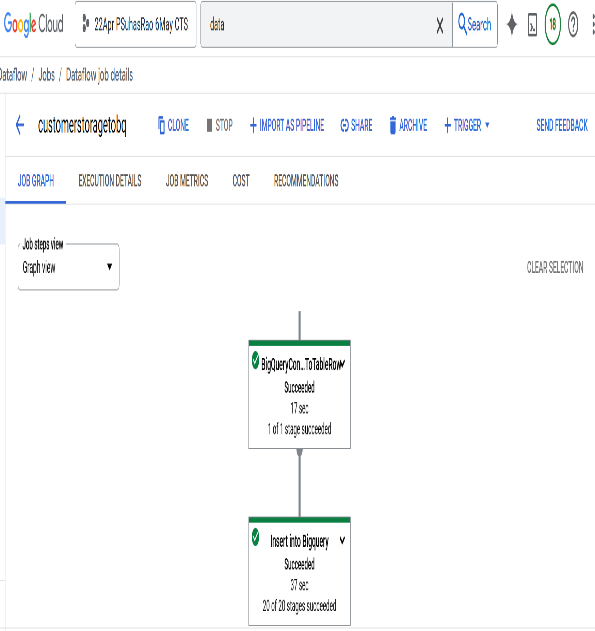
The **Cloud Dataflow "Text Files on Cloud Storage to BigQuery" template** was used to launch Dataflow jobs. These jobs:

* Pulled raw data from Cloud Storage.
* Applied the transformation logic using the .js files.
* Loaded the clean and structured data into BigQuery according to the defined .json schema.

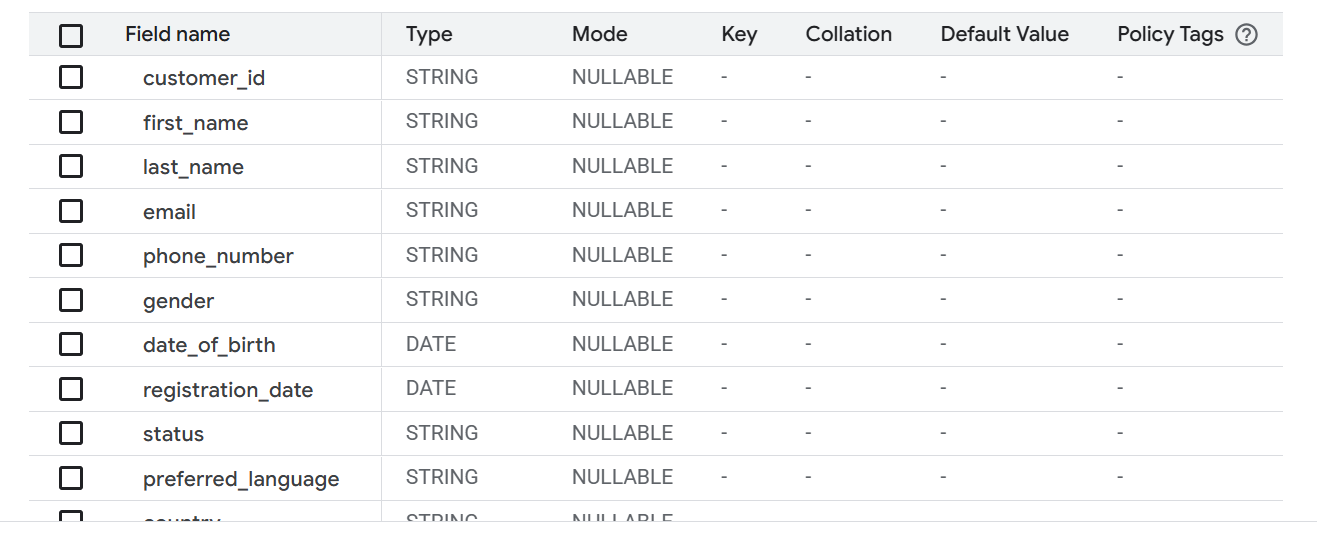
This method effectively mimicked a real-time pipeline using batch jobs, ensuring schema enforcement and preprocessing before storage in BigQuery.

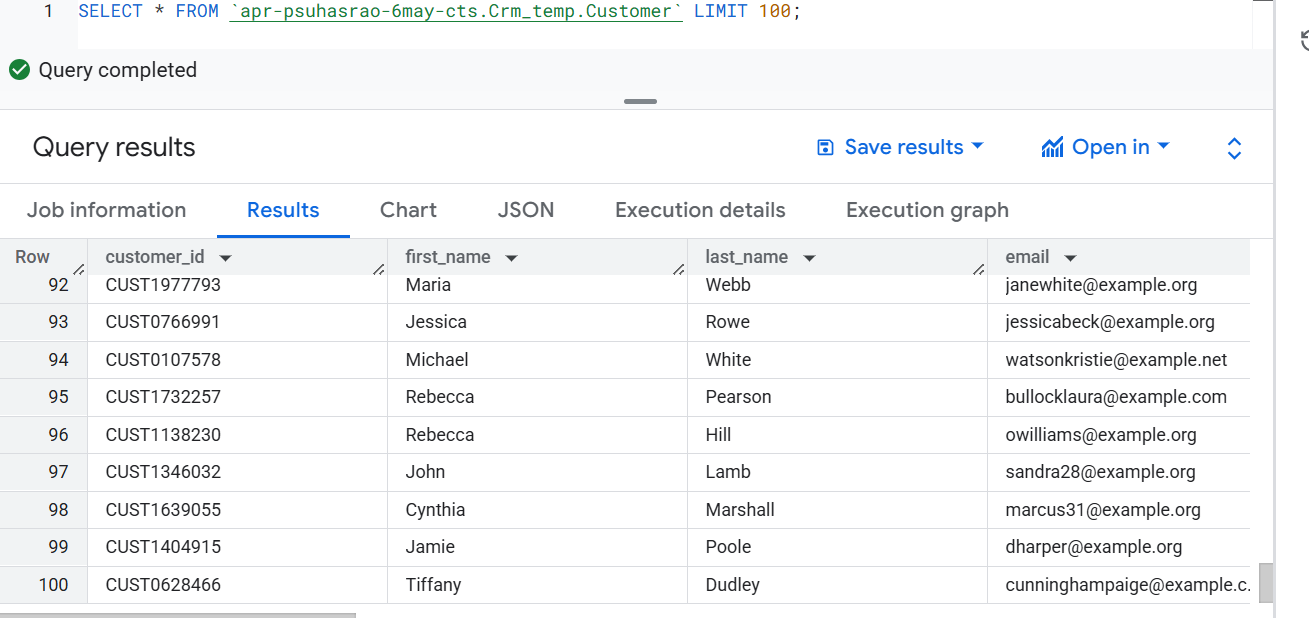
(12) Customer.js (13) Customer.json

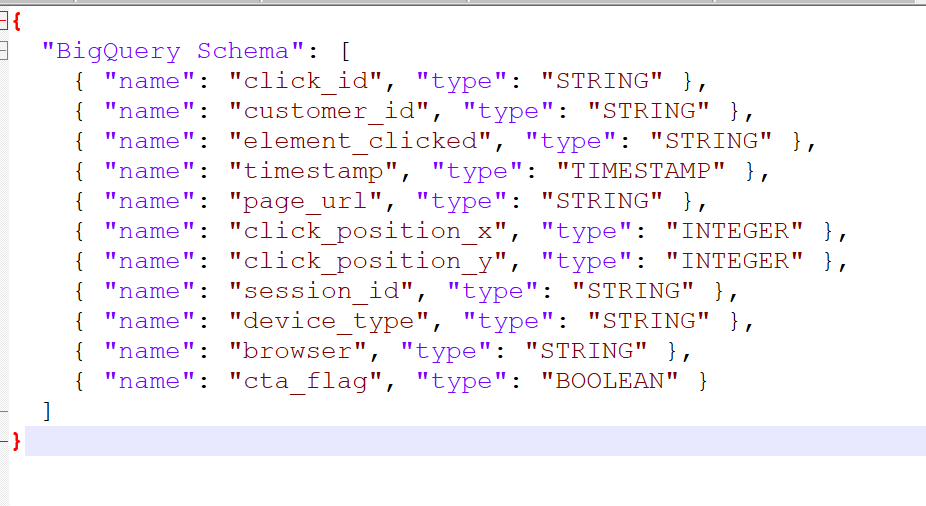
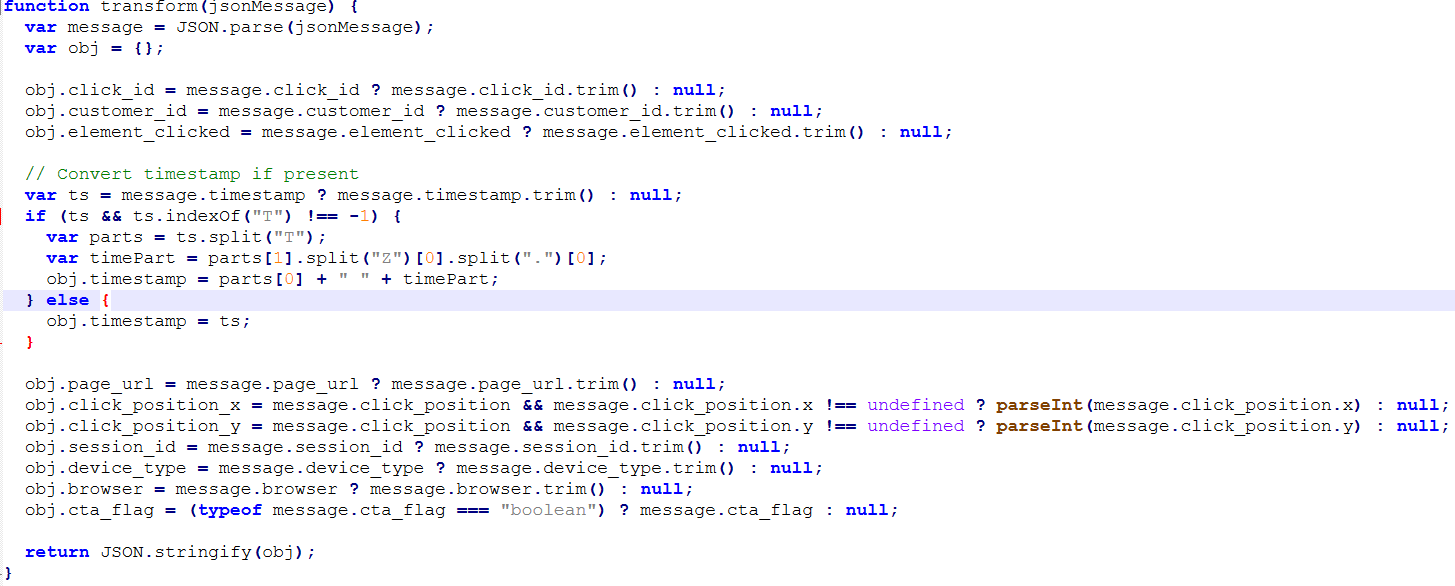
(14) Customer Streaming Dataflow Job



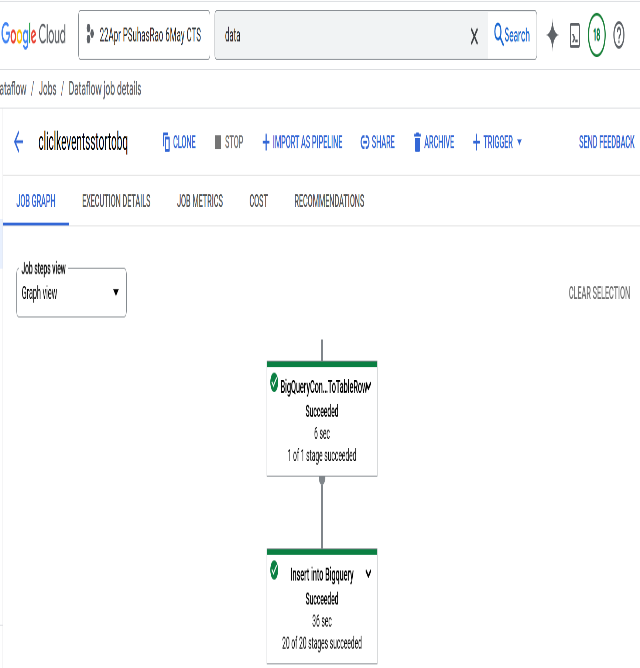
(15) Customer Table Schema



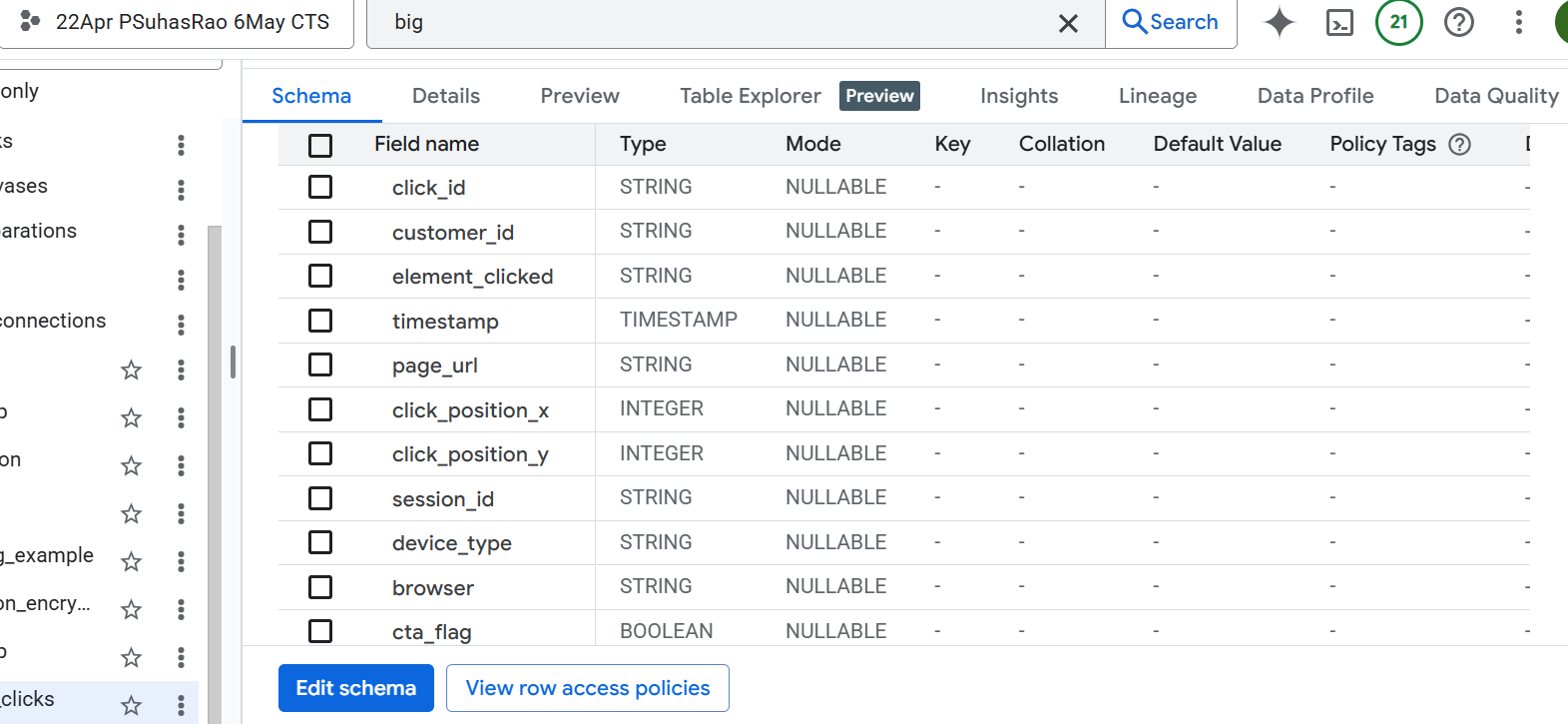
(16) Transformed Customer Data



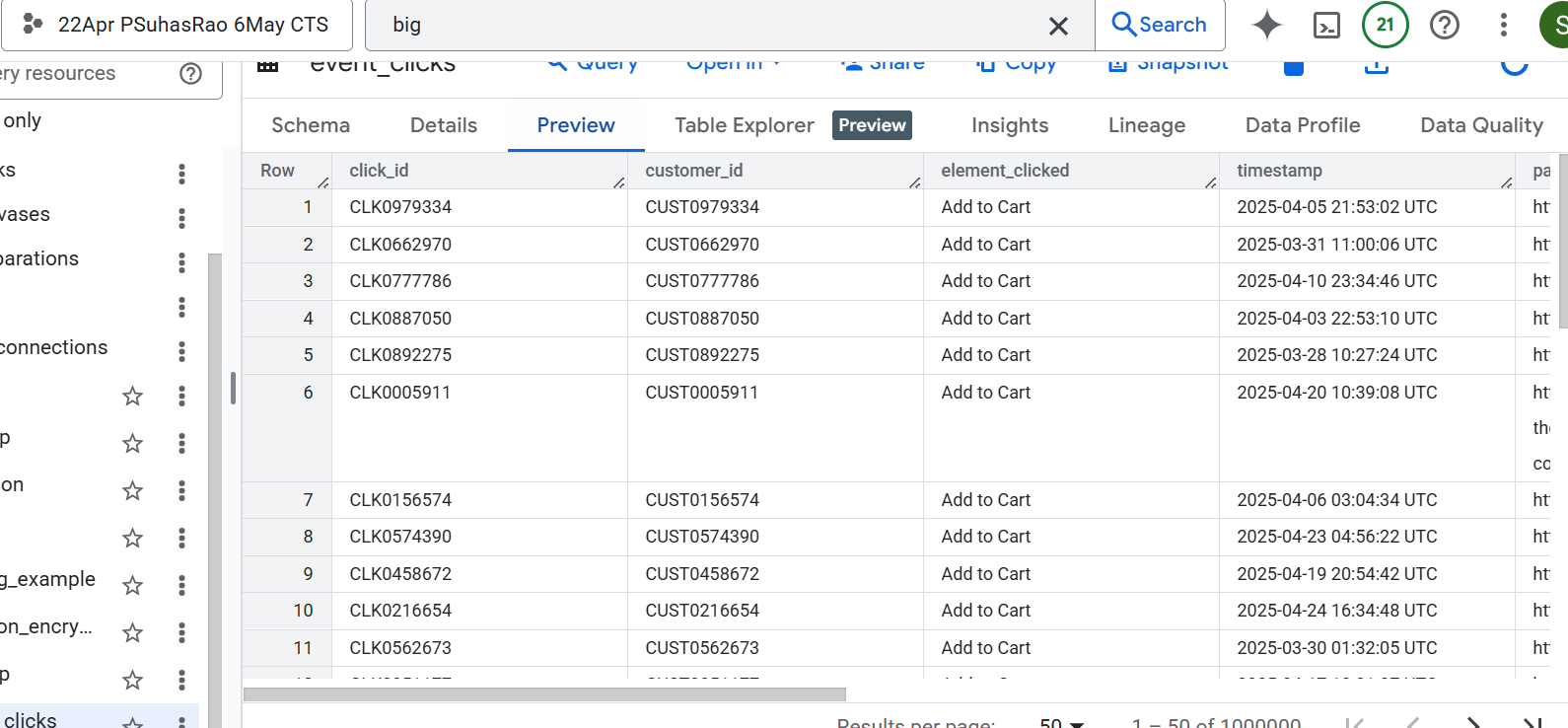
(17) Click\_Events.js (18) Click\_Events.json

(19) Click events Streaming Dataflow Job



(20) Click Events Table Schema



(21) Transformed Click Events Data

Encryption of Sensitive fields in Transactional Data

The following sensitive fields were encrypted before storage:

**orders1**: customer\_id, shipping\_address, billing\_address, payment\_method

**order\_items**: supplier\_id, supplier\_name

**payments1**: customer\_id, payment\_method, transaction\_id, authorization\_code, account\_number\_masked, payment\_ip

**invoices1**: customer\_id, billing\_address, payment\_method, bank\_name, account\_number\_masked

**shipments1**: customer\_id, tracking\_number, origin, destination

**returns1**: customer\_id, return\_reason, processed\_by

**refunds1**: customer\_id, bank\_name, account\_number\_masked, refund\_reason

**inventory**: supplier\_id, batch\_number

**product\_catalog**: supplier\_id

**promotions**: promotion\_code

**payment\_transactions1**: customer\_id, account\_number\_masked, transaction\_reference

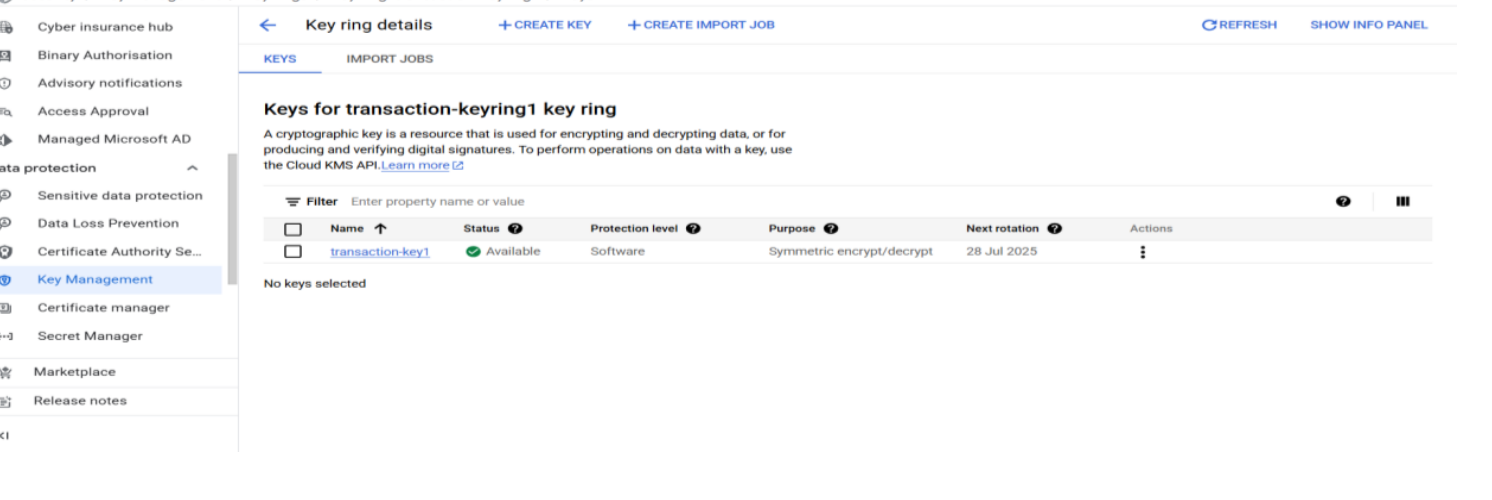
**shipment\_tracking1**: customer\_id, tracking\_number, delivery\_address

**order\_returns1**: customer\_id, pickup\_address

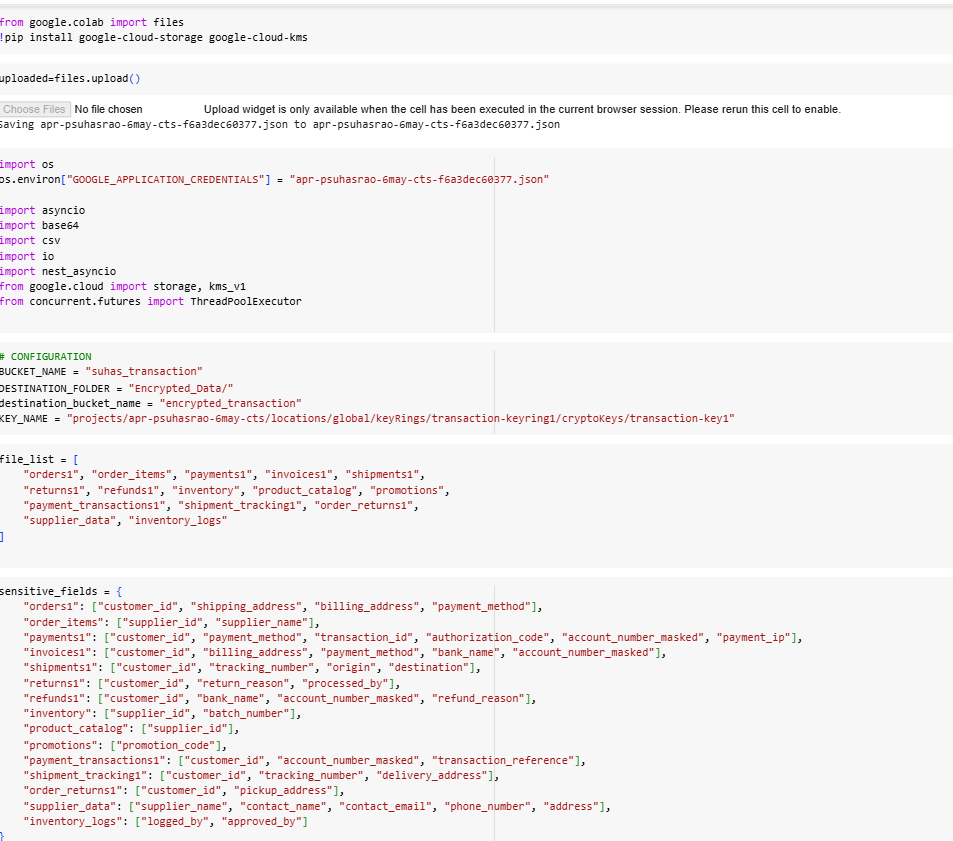
**supplier\_data**: supplier\_name, contact\_name, contact\_email, phone\_number, address

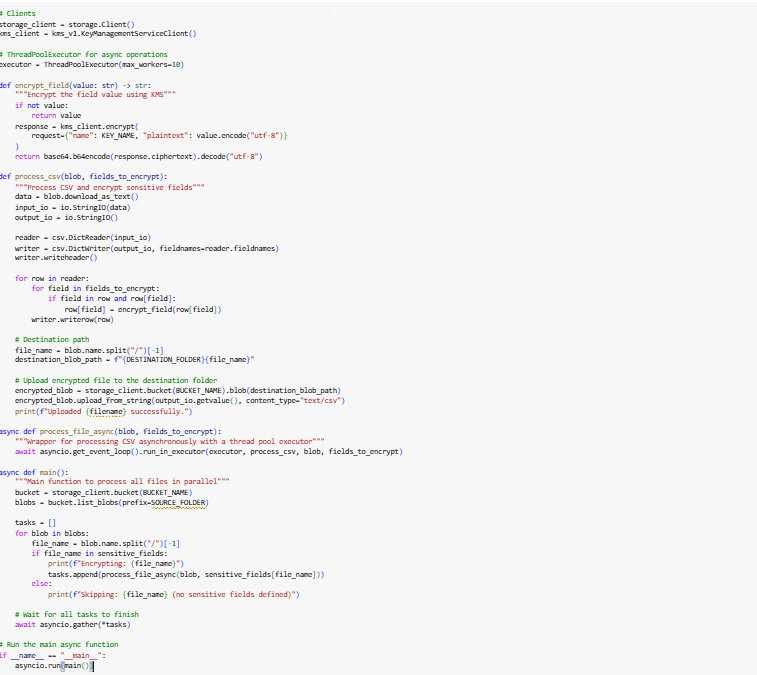
**inventory\_logs**: logged\_by, approved\_by

These fields were identified as containing personally identifiable information (PII) or confidential data and were encrypted to ensure data security during storage and processing.

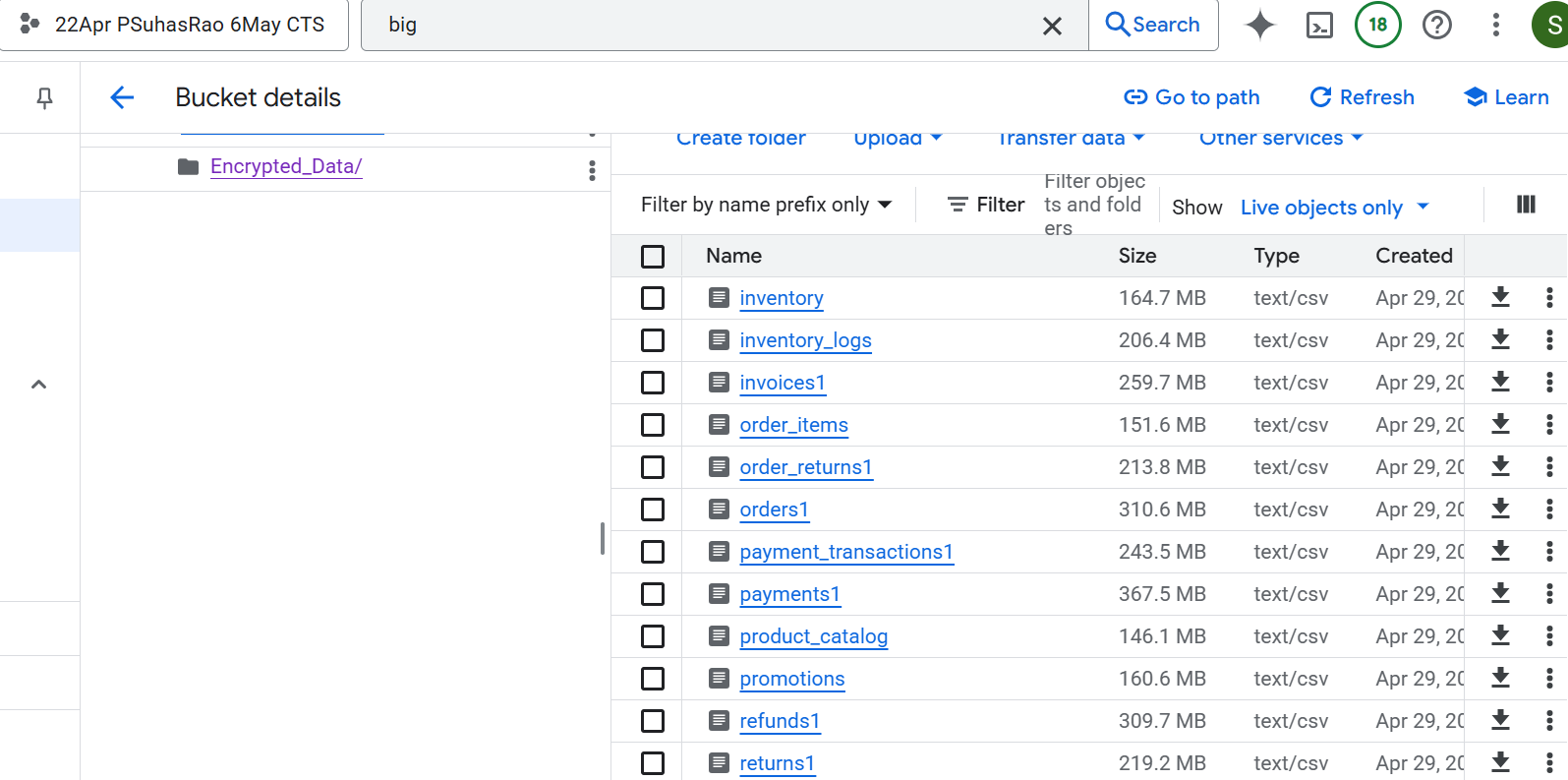


(22) Key Generation in KMS

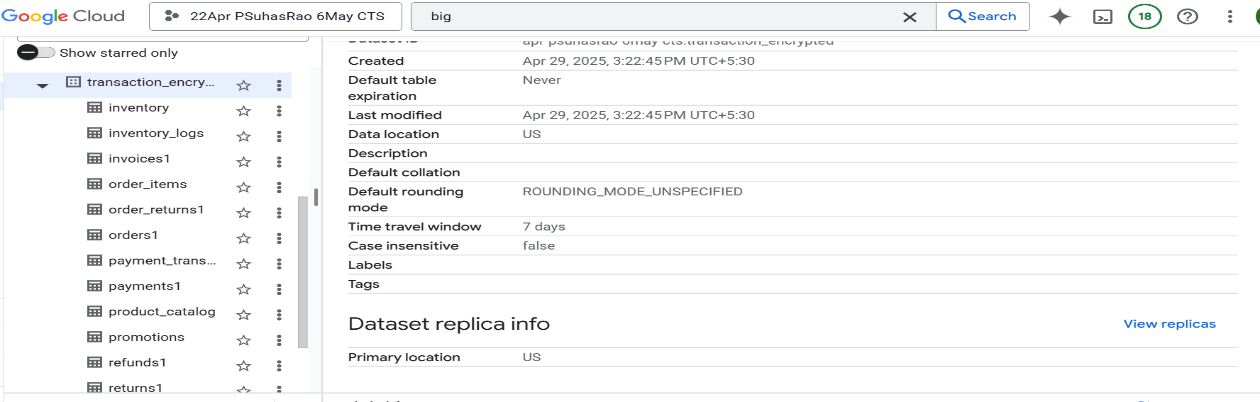




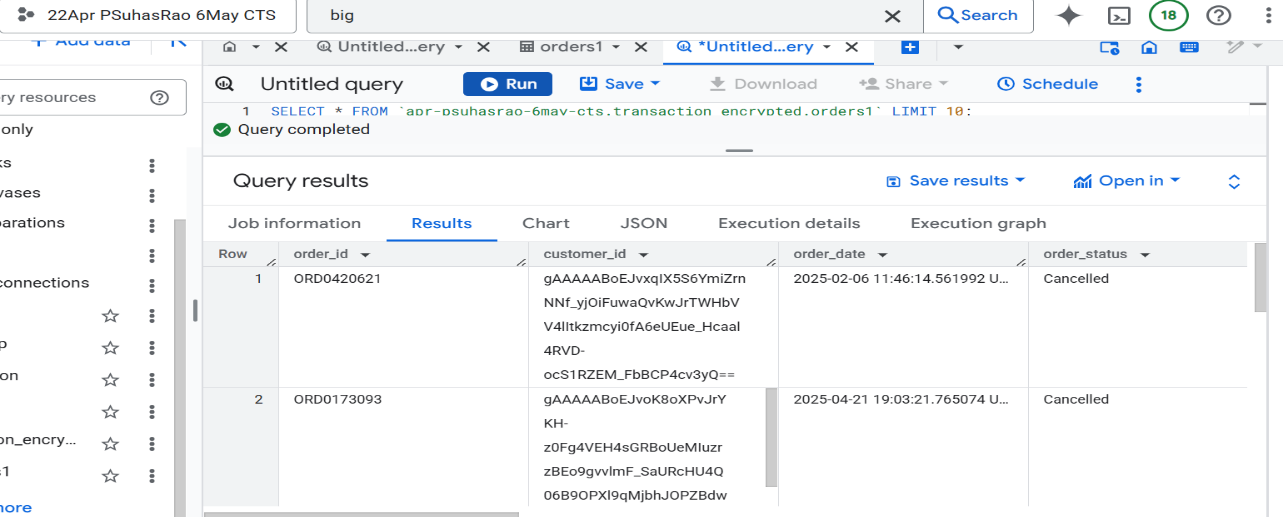
(23) Python Code for Encryption



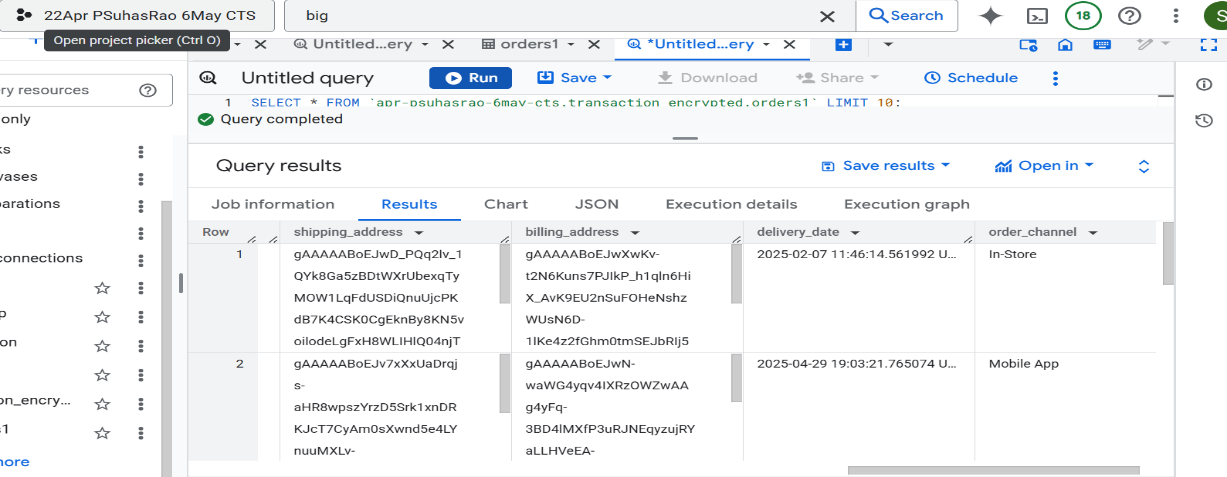
(24) Bucket in which Encrypted Data is stored



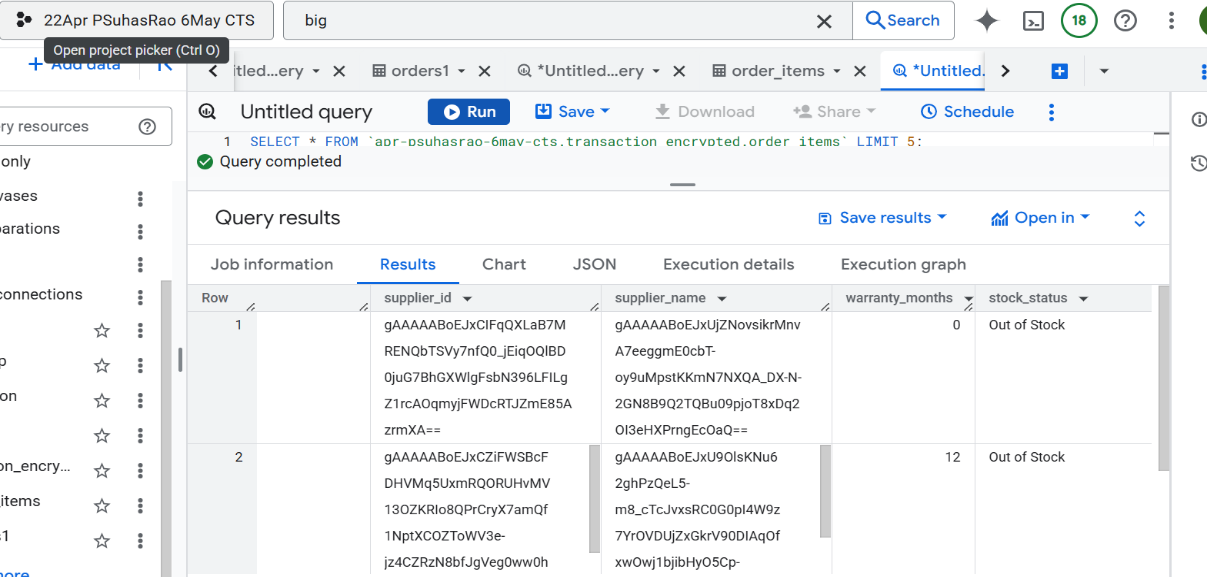
(25) Encrypted Transactional Data in BigQuery



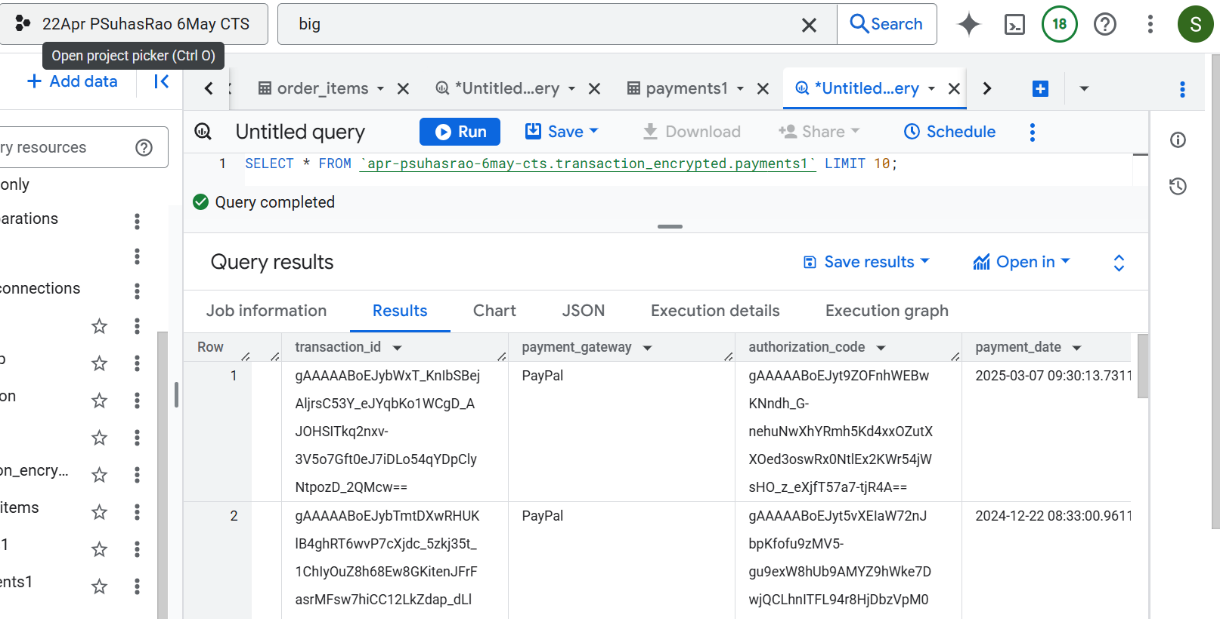
(26) Encrypted Orders Data



(27) Encrypted Orders Data



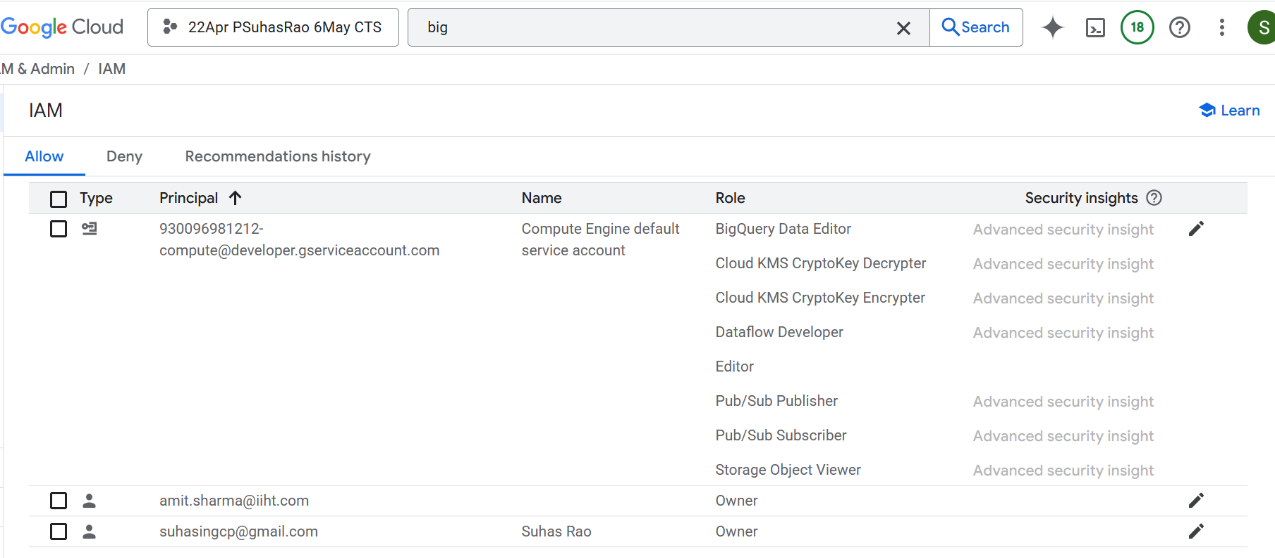
(28) Encrypted Order Items Data



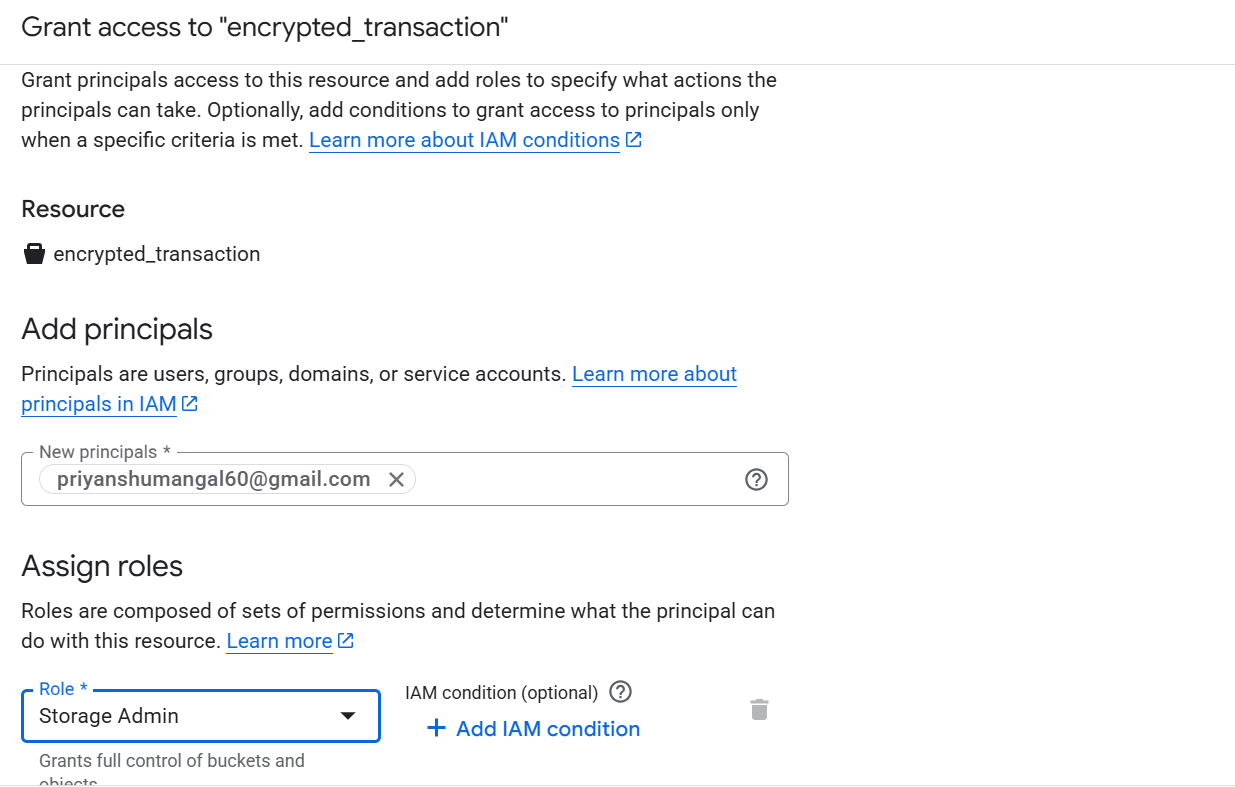
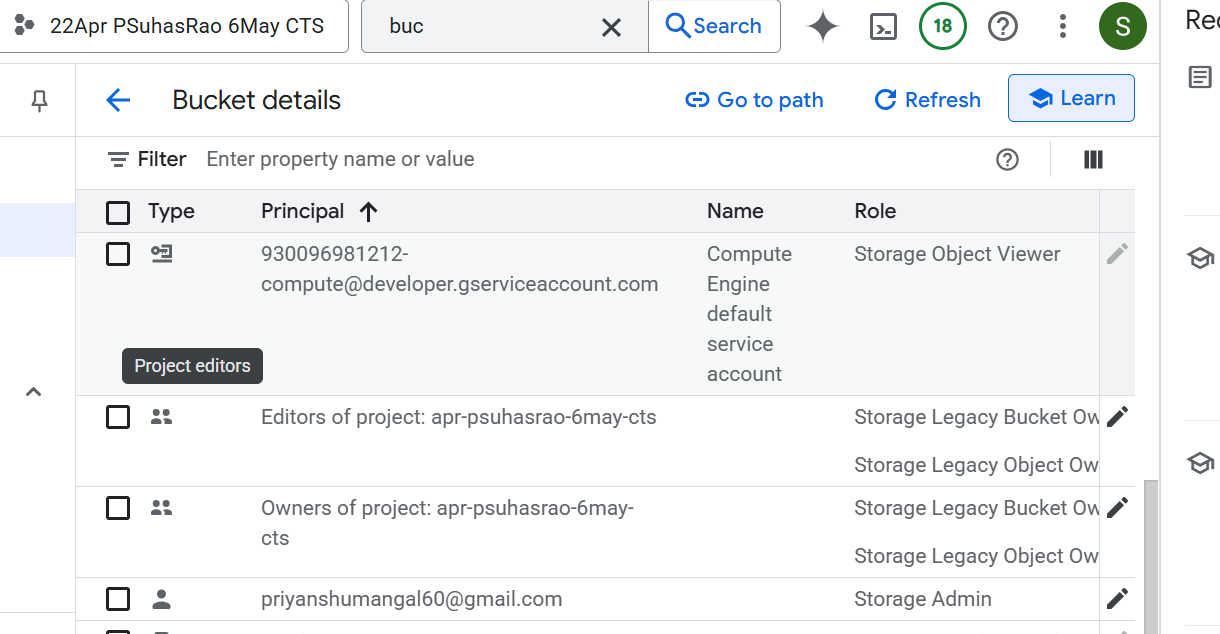
(29) Encrypted Payments Data

IAM and VPC

We were not granted permission to access the VPC at the organizational level.

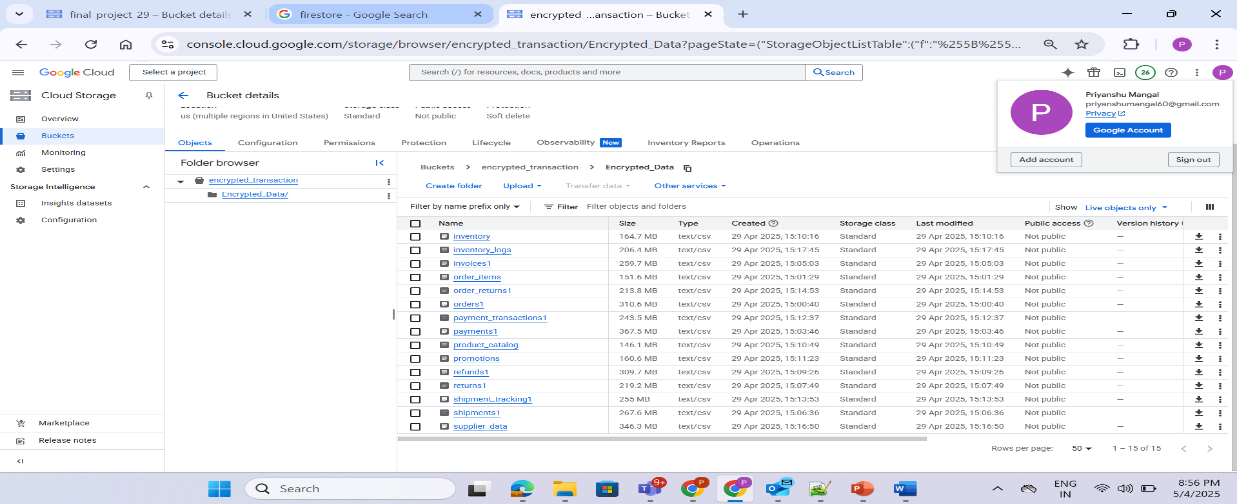


(30) Permissions Provided for Service Account

(31) Storage Admin Role

Provided [priyanshumangal60@gmail.com](mailto:priyanshumangal60@gmail.com) a role of storage admin for the bucket encrypted\_transaction so that he can access the sensitive data.



(32) Bucket Accessed By [priyanshumangal60@gmail.com](mailto:priyanshumangal60@gmail.com)

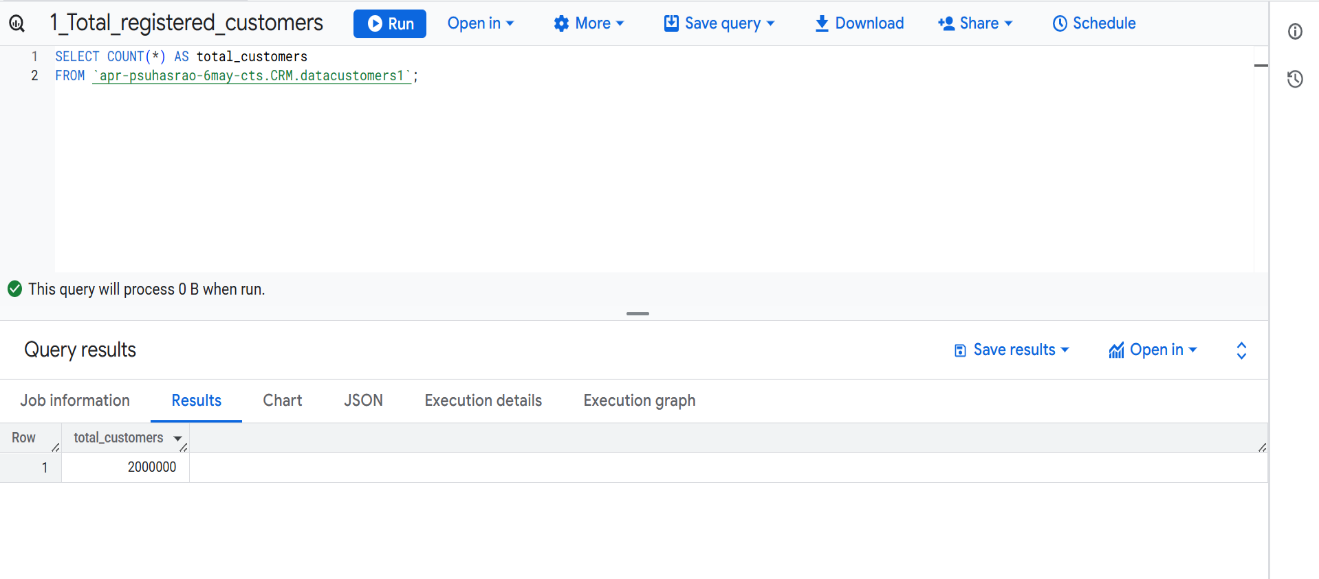
Chapter 5:

**Results and Discussions**

To evaluate the effectiveness of the implemented Customer Data Platform and to derive actionable insights from the processed data, a set of business-critical Key Performance Indicators (KPIs) were defined and visualized through interactive dashboards. These metrics and visualizations provide a comprehensive view of customer behavior, engagement trends, and transactional performance.

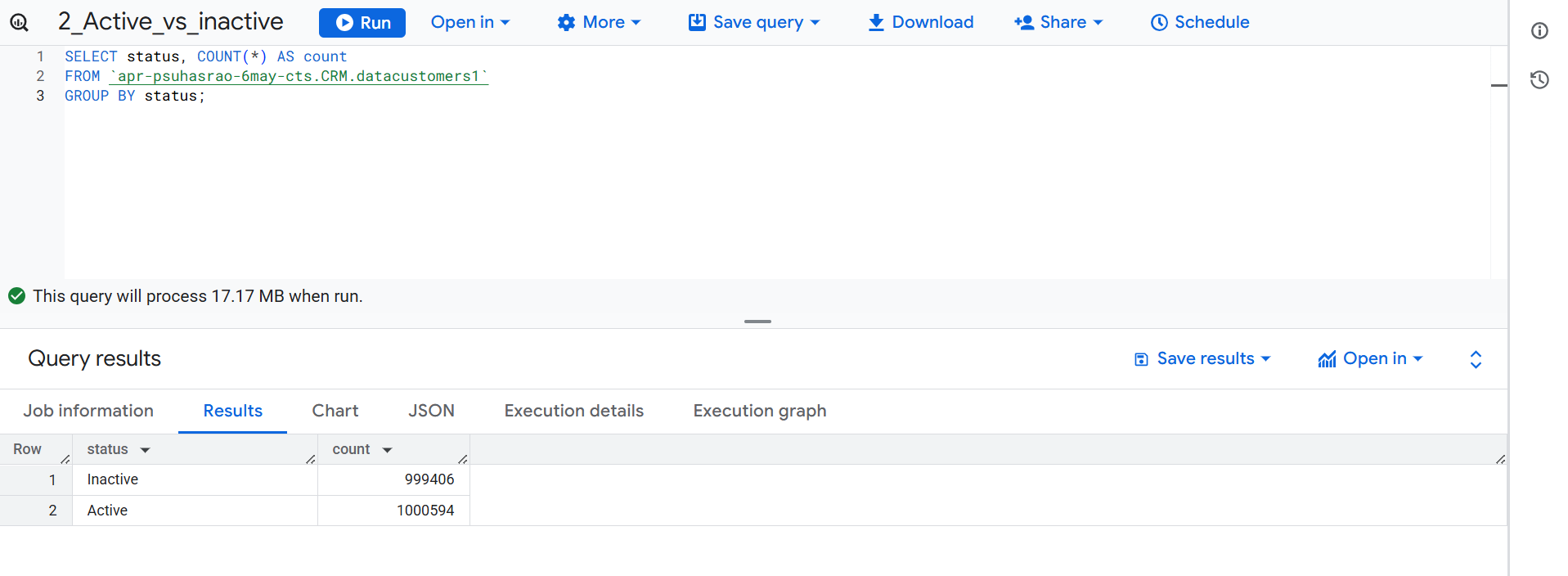
Key Performance Indicators (KPI’s)

**KPI 1: Total Registered Customers**



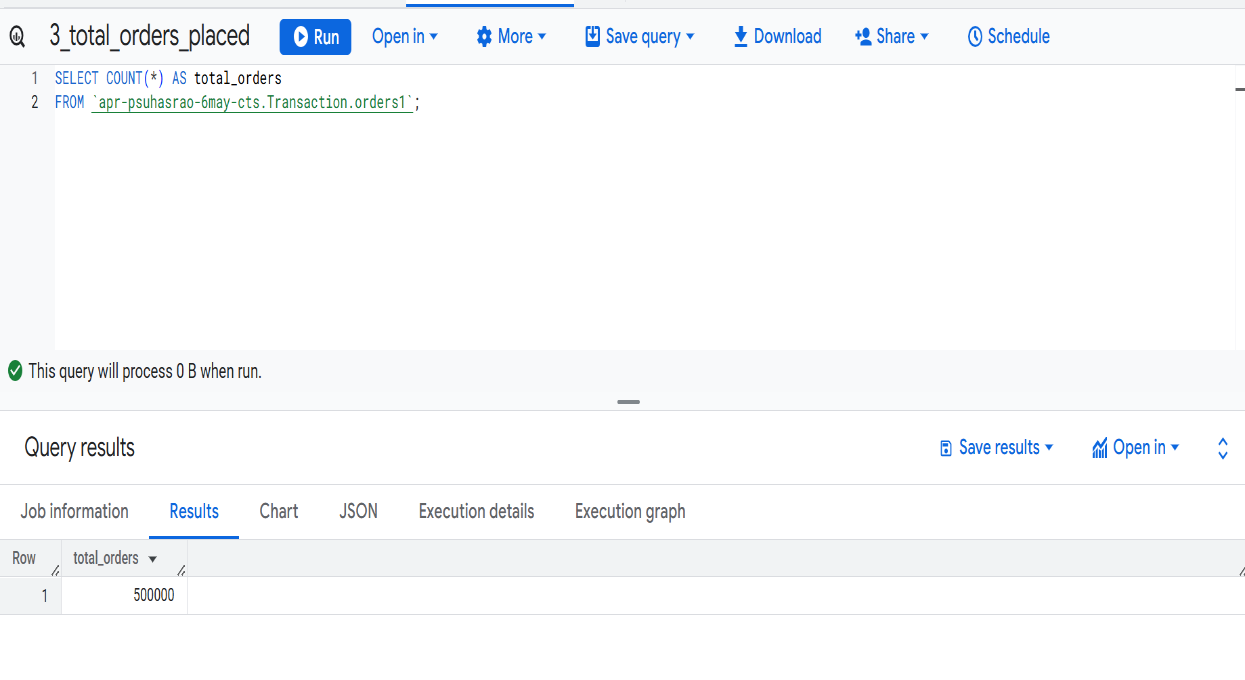
(33) KPI 1

**KPI 2: Active vs Inactive Customers**

****

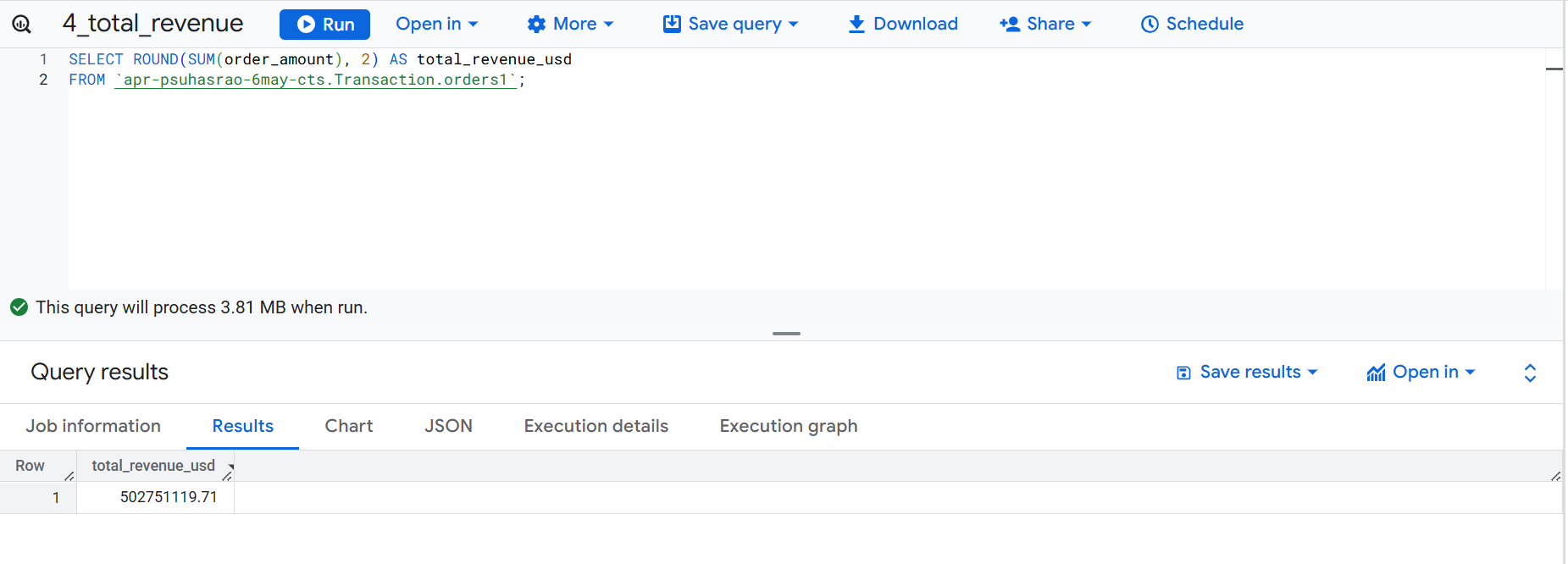
(34) KPI 2

**KPI 3: Total Orders Placed**

****

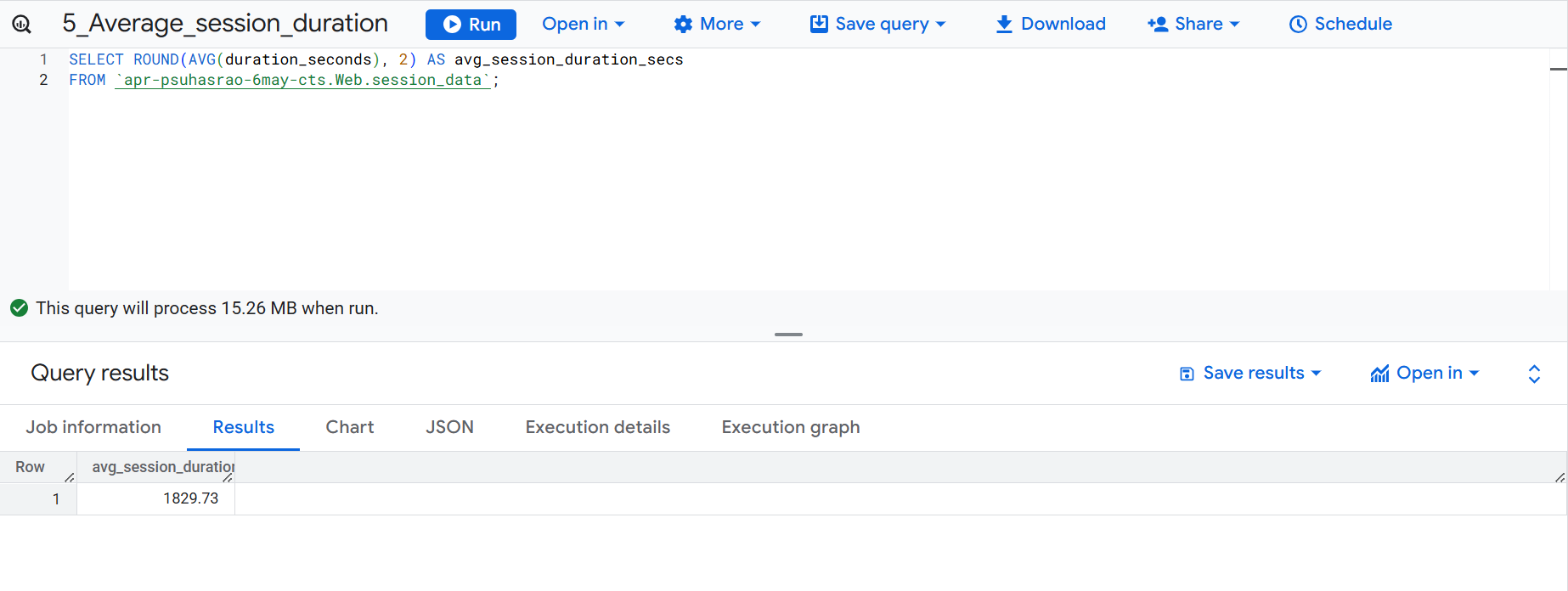
(35) KPI 3

**KPI 4: Total Revenue**

****

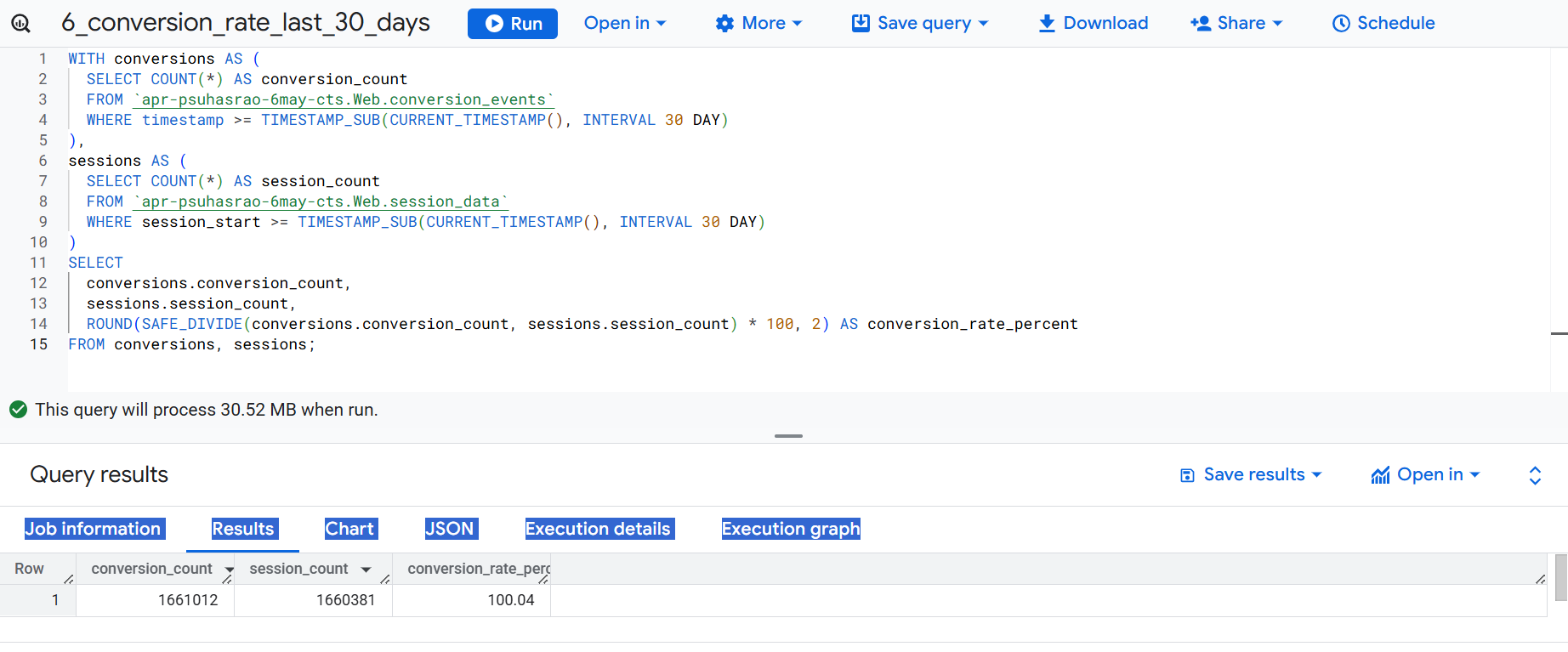
(36) KPI 4

**KPI 5: Average Session Duration**

****

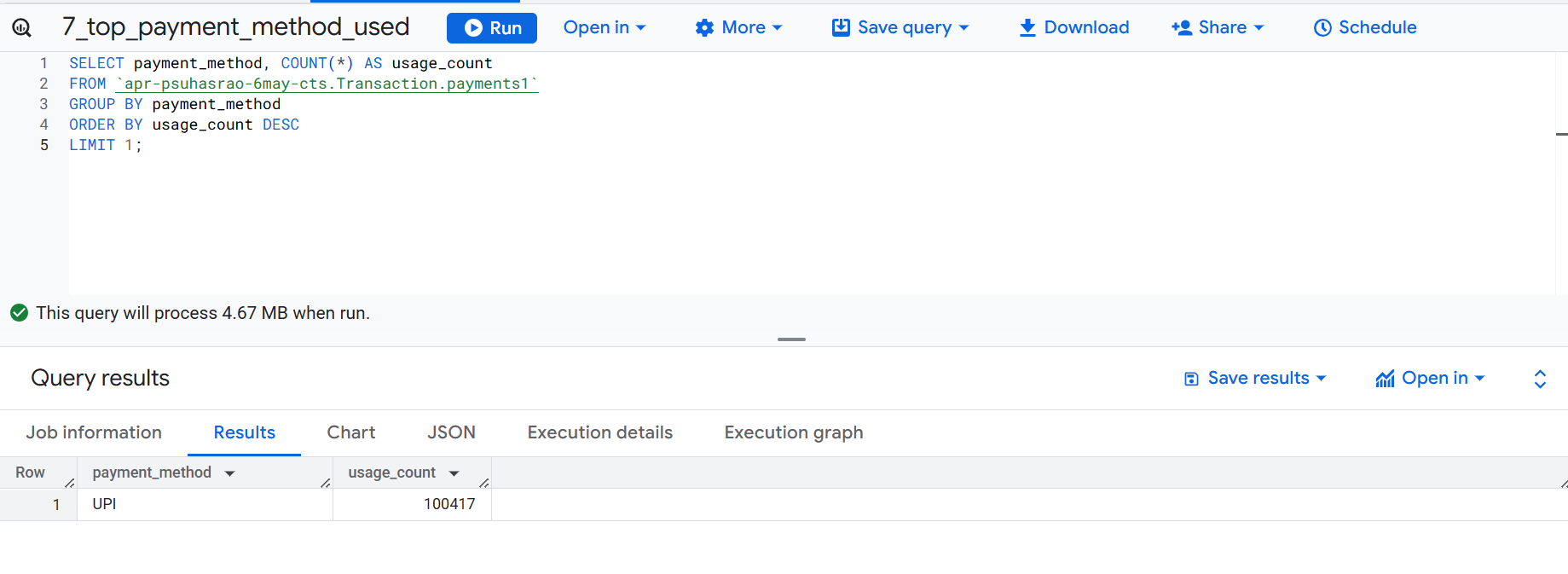
(37) KPI 5

**KPI 6: Conversion Rate (Last 30 Days)**

****

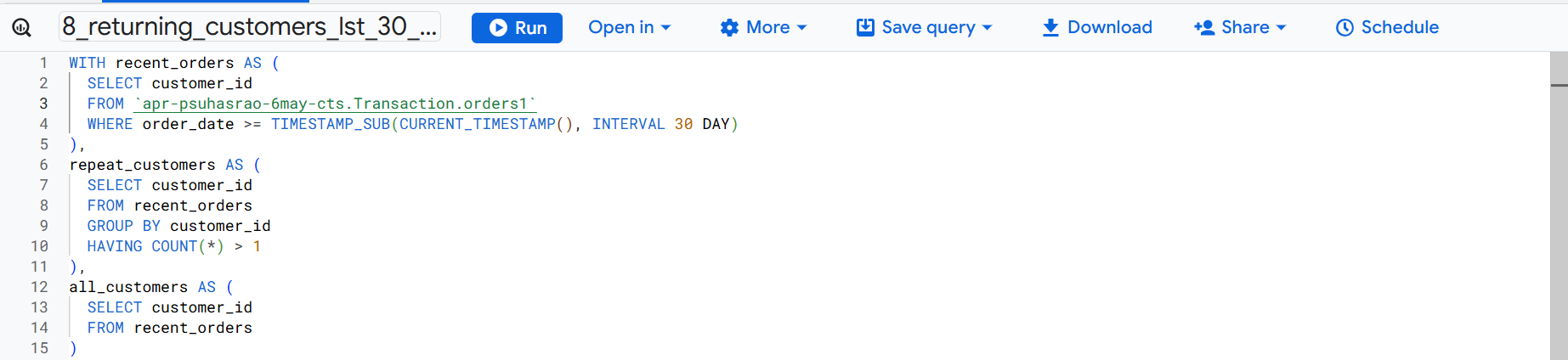
(38) KPI 6

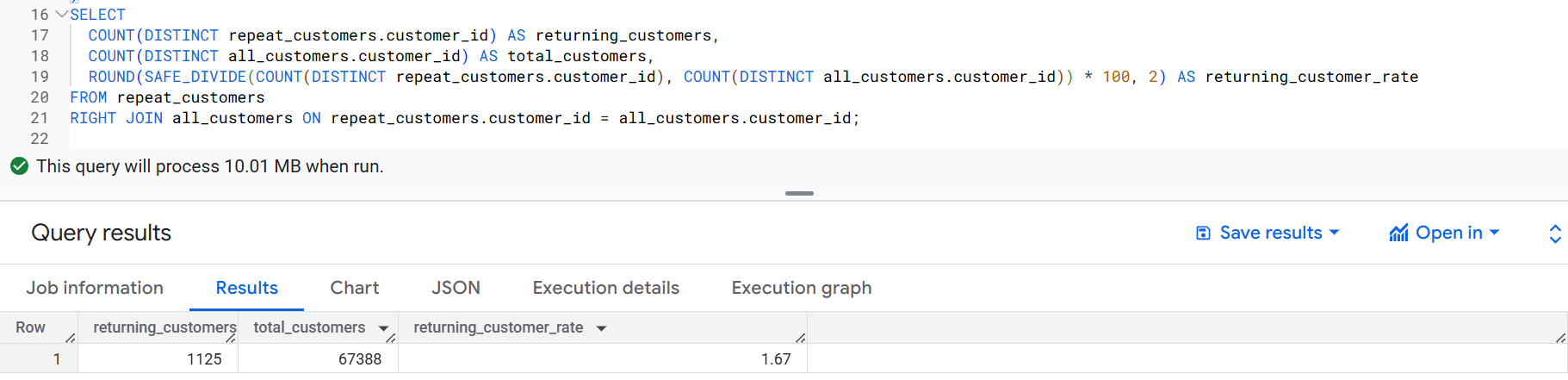
**KPI 7: Top Payment Method Used**

****

(39) KPI 7

**KPI 8: Returning Customer Rate (Last 30 Days)**

****

****

(40) KPI 8

**Dashboards**

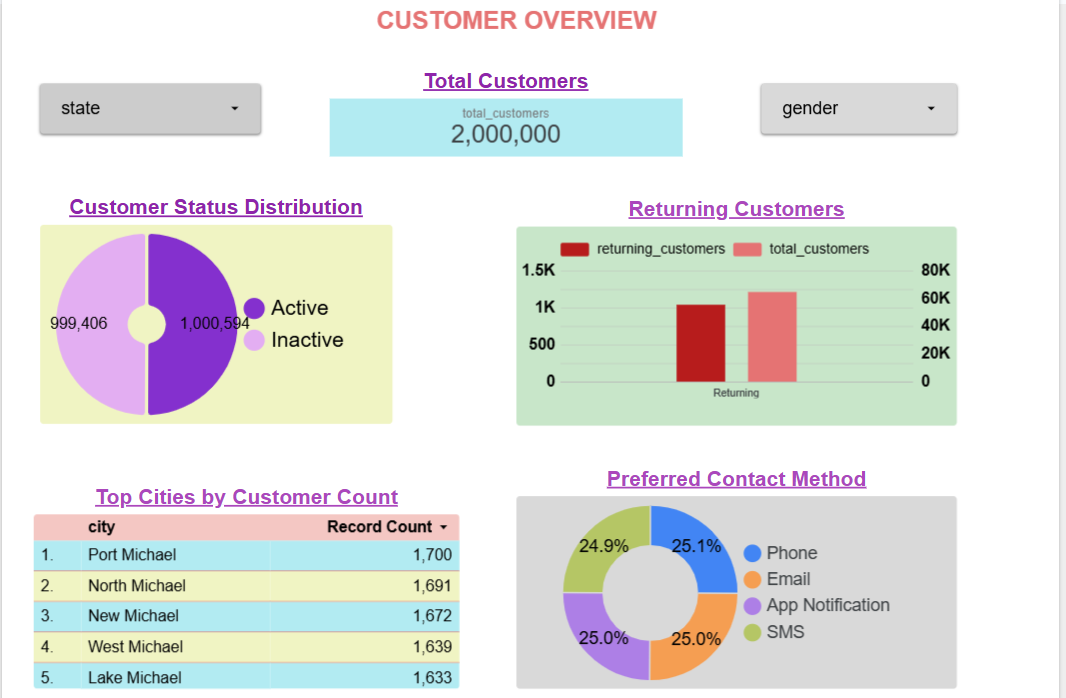
To visualize the processed data effectively, three dashboards were developed:

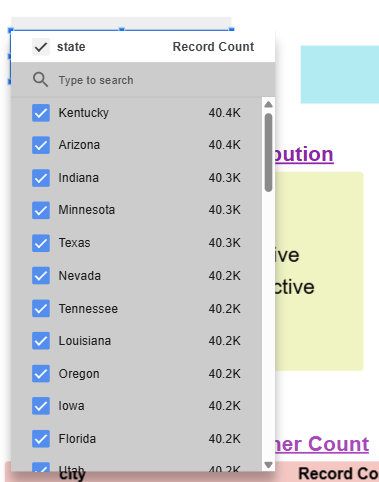
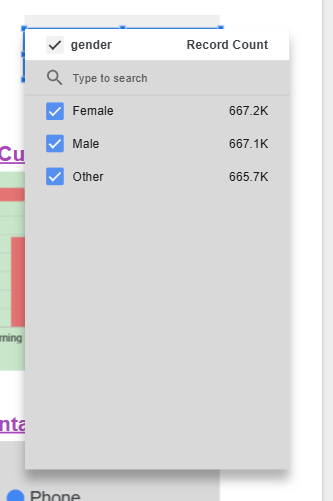
**Customer Overview**: Displays key customer metrics including total customers, active users, and customer segmentation.

**Web Engagement Analytics**: Highlights user interaction patterns such as page views, click events, browsers used etc.

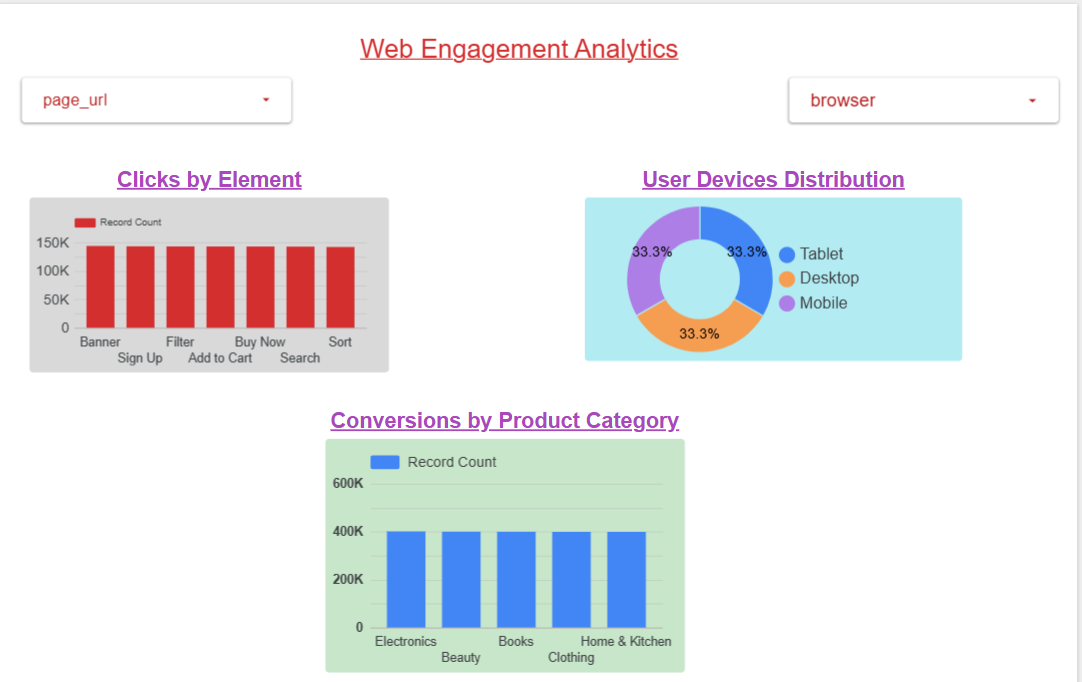
**Transaction Insights**: Provides visibility into order volumes, payment modes, and revenue trends to support financial analysis.

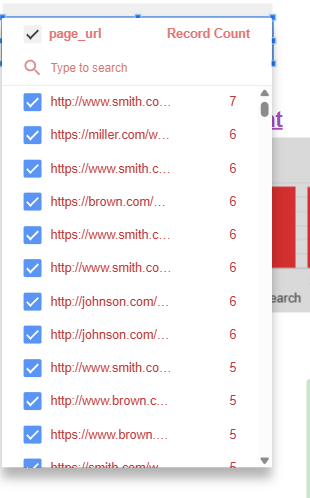
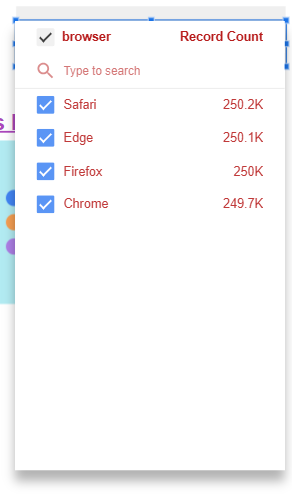
These dashboards help monitor performance and support data-driven decisions.



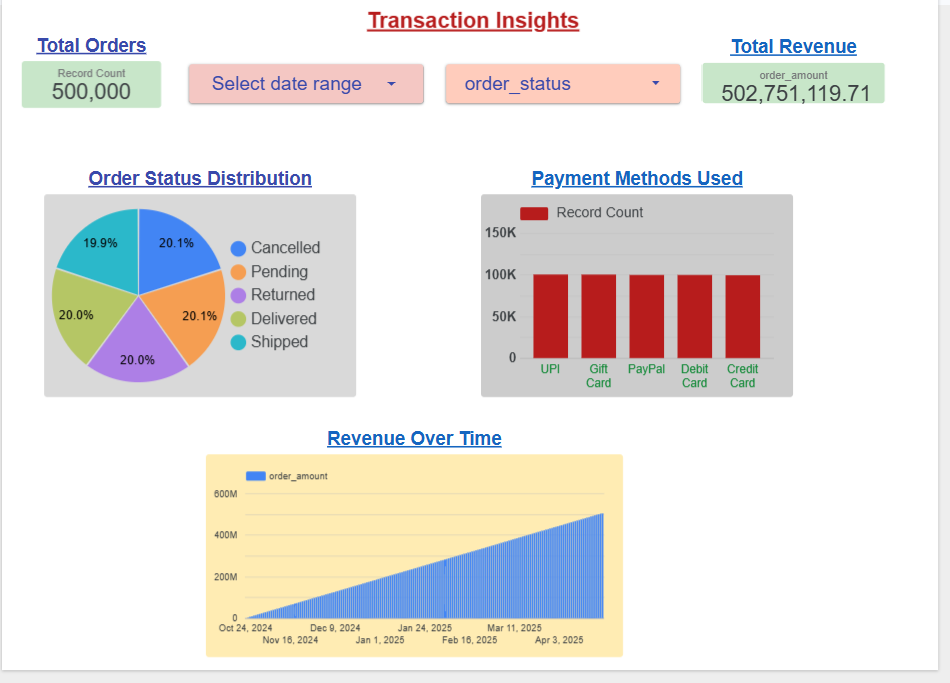
 

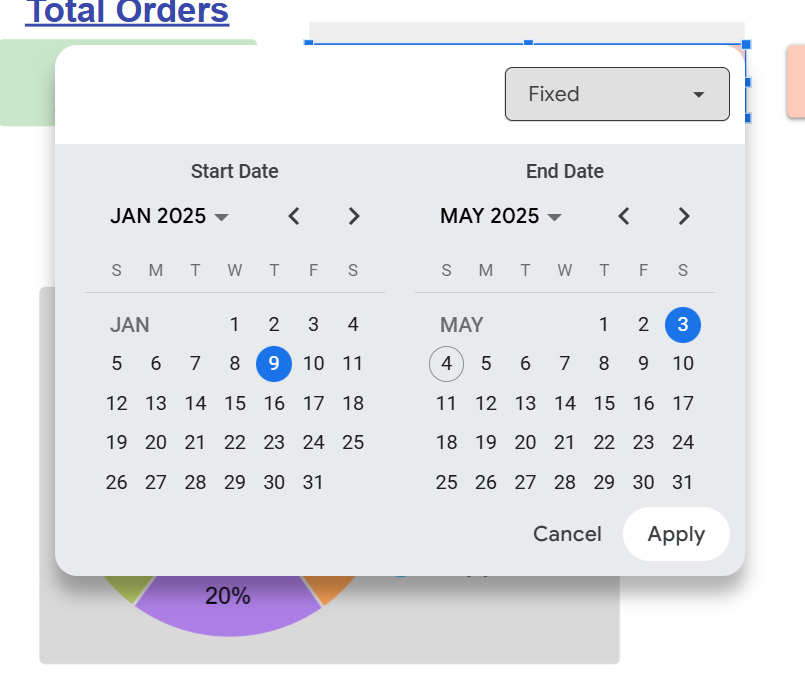
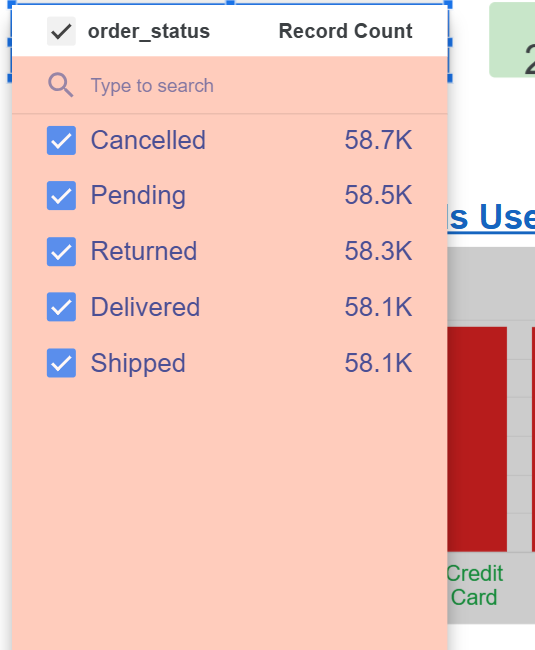
(41) Customer Overview Dashboard



(42) Web Analytics Dashboard

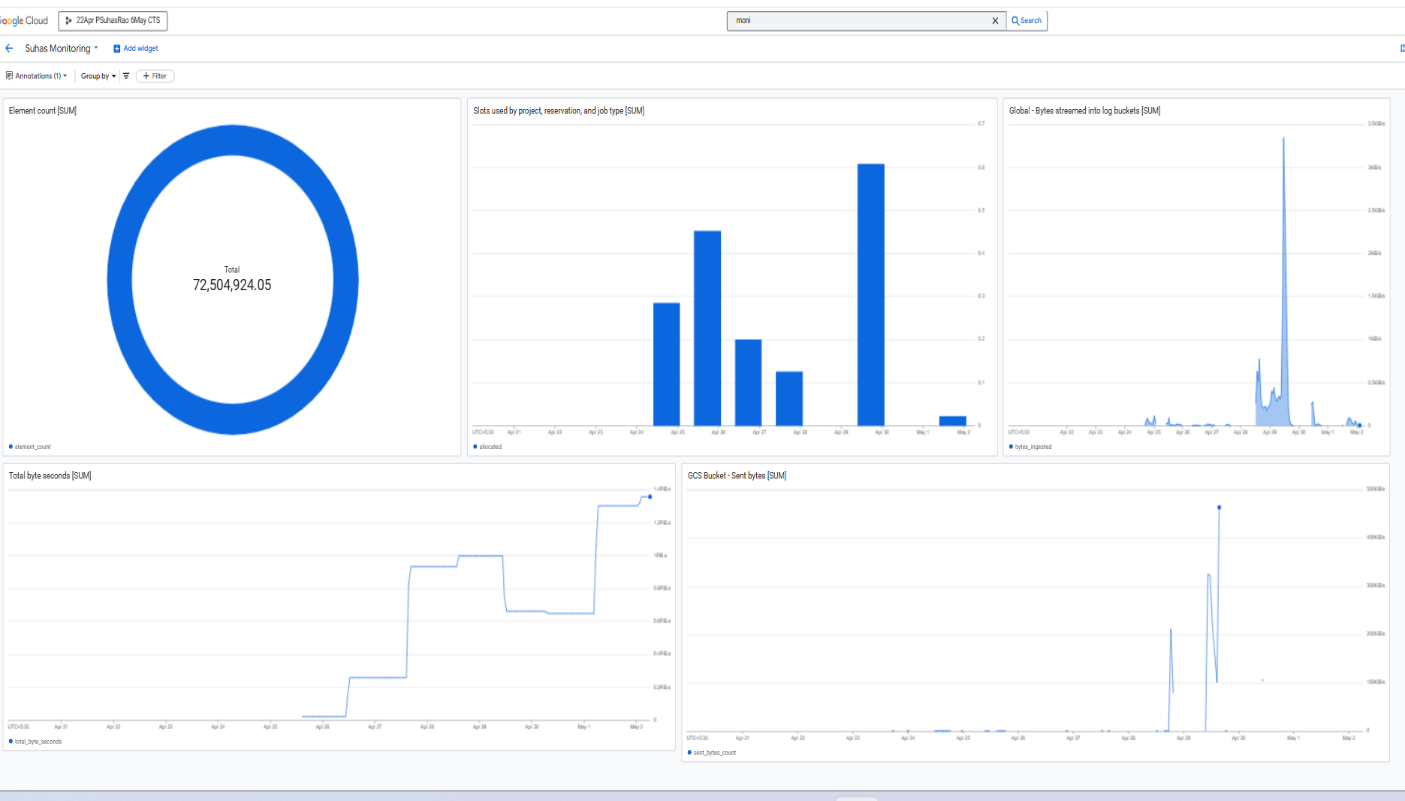


(43) Transaction Insights Dashboard

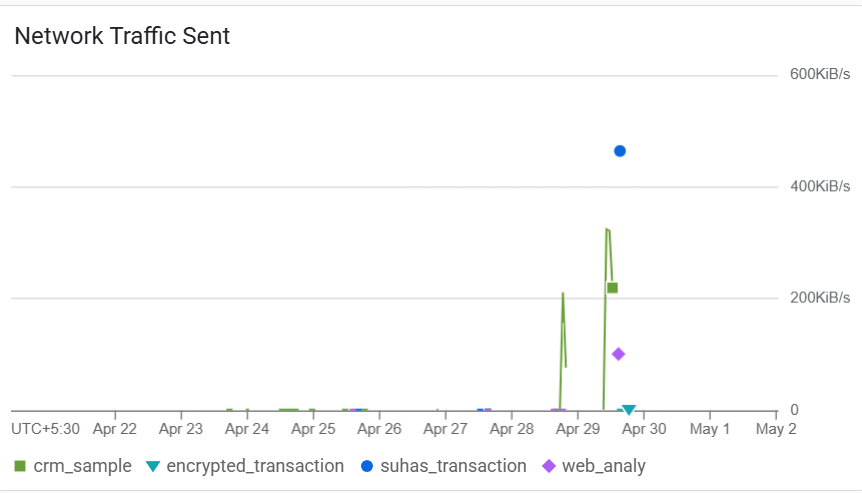
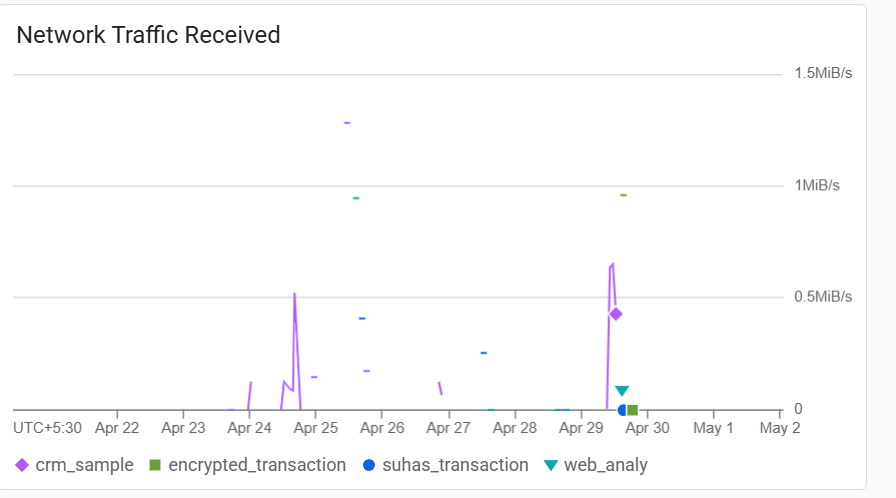
**Monitoring and Logging**

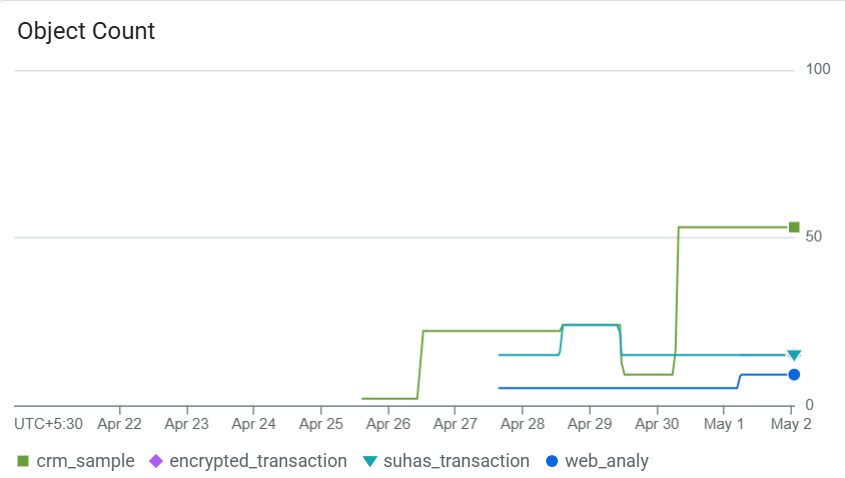
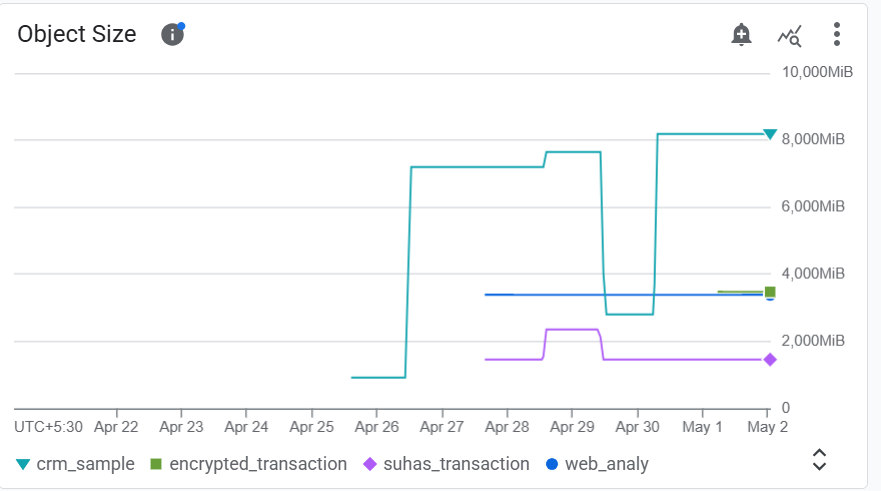
* Use of Google Cloud Monitoring to track BigQuery job errors.
* Alerting mechanisms (e.g., email alerts when errors occur).
* Logging dashboards to monitor data ingestion and processing pipelines.



(44) Monitoring Dashboard

**Cloud Storage Monitoring**

**** 

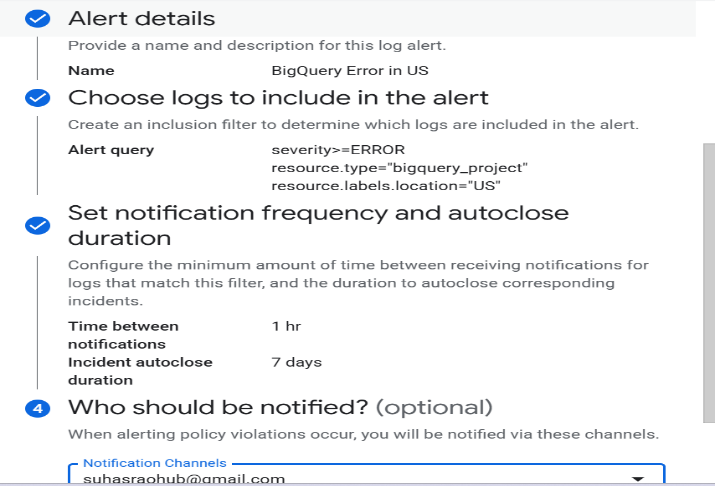
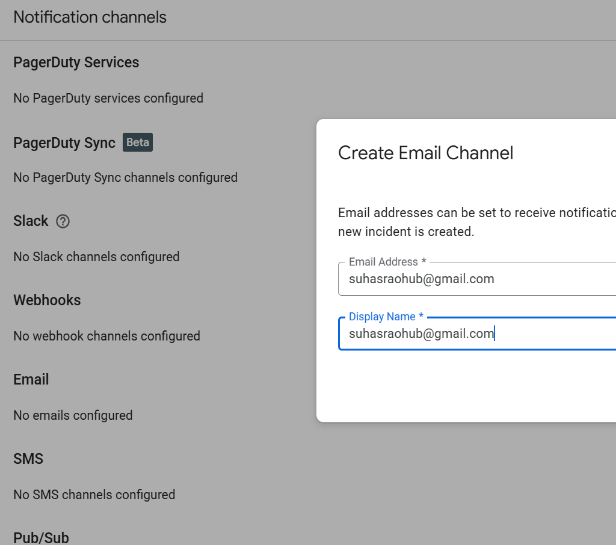
****

(45) Cloud Storage Monitoring

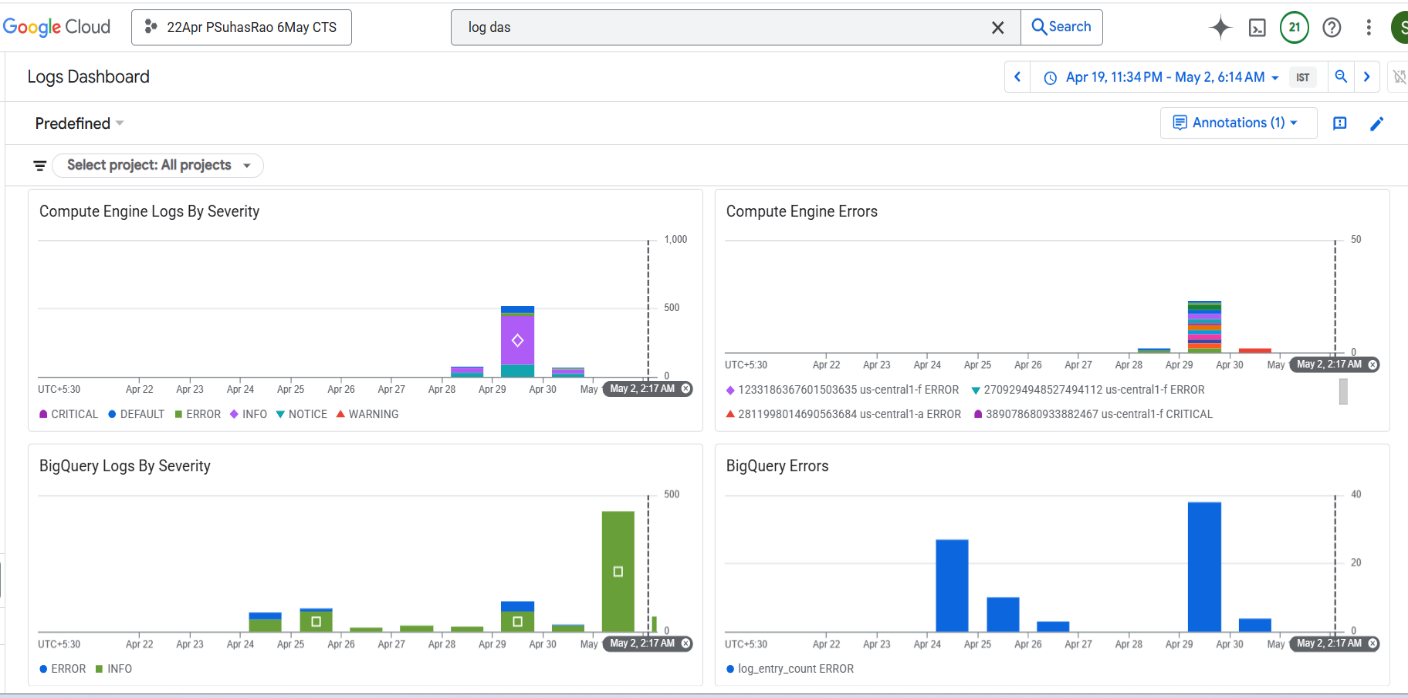
An alert will be sent to my email address [suhasraohub@gmail.com](mailto:suhasraohub@gmail.com) whenever error is triggered in bigquery project.

A screenshot of a computer

AI-generated content may be incorrect.

(46) BigQuery Error Alerting



(47) Logging Dashboard

Chapter 6:

# Conclusion and Future Scope

**Conclusion**

The implementation of the Customer Data Platform (CDP) on Google Cloud Platform (GCP) successfully demonstrated the ability to ingest, process, secure, and visualize large volumes of customer and web analytics data in real time. By leveraging GCP-native services such as Pub/Sub, Dataflow, BigQuery, Cloud KMS, IAM, and Looker Studio, the platform ensures scalability, security, and operational efficiency.

The project achieved its core objectives, including:

* Real-time and batch data ingestion from CRM, web, and transactional sources.
* Secure storage and encryption of sensitive data fields.
* Computation of key business metrics (KPIs).
* Visualization of insights through interactive dashboards.
* Implementation of monitoring and alerting mechanisms for system reliability.

This platform provides a strong foundation for data-driven decision-making and enhances data governance across business functions.

**Future Scope**

While the current implementation meets essential business requirements, there are several opportunities to enhance the platform further:

* **Integration of Machine Learning Models**: Incorporating predictive analytics for customer churn, segmentation, and recommendation systems.
* **Third-Party Tool Integration**: Connecting with external marketing platforms like Salesforce, HubSpot, or Google Ads for campaign automation.
* **Real-Time Personalization**: Using real-time data to personalize user experiences across digital platforms.
* **Scalability Improvements**: Optimizing pipelines for larger datasets and higher throughput.

Chapter 7:

**References**

* Google Cloud Platform Documentation – <https://cloud.google.com/docs>
* Pub/Sub Documentation – <https://cloud.google.com/pubsub/docs>
* Dataflow Documentation – <https://cloud.google.com/dataflow/docs>
* BigQuery Documentation – <https://cloud.google.com/bigquery/docs>
* Cloud KMS Documentation – <https://cloud.google.com/kms/docs>
* IAM Documentation – <https://cloud.google.com/iam/docs>
* Looker Studio Documentation – <https://cloud.google.com/looker/docs>
* Cloud Monitoring and Logging – <https://cloud.google.com/monitoring/docs>
* Faker Library for Python – <https://faker.readthedocs.io>