Project Report

On

Real-time Ecommerce Analytics Dashboard

Submitted

in the partial fulfillment for the award of

Post Graduate Diploma in Big Data Analytics (PG-DBDA)

from **Know-IT ATC, CDAC ACTS, Pune**

Guided by:

Mr. Anay Tamhankar & Mr. Prasad Deshmukh

Submitted By:

Harshal Amrutkar	(230343025005)
Vedant Bhosale	(230343025010)
Akshay Funde	(230343025012)
Vijay Munde	(230343025034)

CERTIFICATE

<u>TO WHOMSOEVER IT MAY CONCERN</u>

This is to certify that.

Harshal Amrutkar (230343025005)

Vedant Bhosale (230343025010)

Akshay Funde (230343025012)

Vijay Munde (230343025034)

Have successfully completed their project on

Real-time Ecommerce Analytics Dashboard

Under the guidance of

Mr. Anay Tamhankar

&

Mr. Prasad Deshmukh

ACKNOWLEDGEMENT

This project **“Real-time Ecommerce Analytics Dashboard”** was a great learning experience for us, and we are submitting this work to Know-IT ATC, CDAC CTS, Pune.

We all are very glad to mention the name of **Mr. Anay Tamhankar & Mr. Prasad Deshmukh** for their valuable guidance at work on this project. Their continuous guidance and support helped us to overcome various obstacles during project work.

We are highly grateful to **Mr. Vaibhav Inamdar** (Center Coordinator, Know-it, Pune) for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data Analytics (PG-DBDA) through C-DAC ACTS, Pune.

Our most heartfelt thanks go to **Mr. Shrinivas Jadhav** (Vice-President, Know-it, Pune) who provided all the required support and his kind coordination to provide all the necessities like required hardware, internet facility and extra lab hours to complete the project, throughout the course and till date, here in Know-IT ATC, CDAC ACTS, Pune.

From:

Harshal Amrutkar	(230343025005)
Vedant Bhosale	(230343025010)
Akshay Funde	(230343025012)
Vijay Munde	(230343025034)

TABLE OF CONTENTS

ABSTRACT

This paper proposes a Real-time Ecommerce Analytics Dashboard using Tableau visualization. In the rapidly evolving landscape of E-commerce, staying ahead requires leveraging real-time data insights.

This project introduces a Real-Time E-commerce Analytics Dashboard, which harnesses the power of Kafka's live data streaming and Tableau's interactive visualization capabilities. Focused on the Olist Public Dataset (a Brazilian E-commerce public dataset), this dashboard transforms raw data into actionable insights, enhancing analytical capabilities.

Our toolkit comprising Python, Apache Spark, ETL, Apache Kafka, MongoDB, and Tableau, ensures seamless data processing and visualization.

The dashboard empowers businesses to grasp customer trends, seller dynamics, and product performance in real time. By continuously processing live data streams through Kafka, the dashboard enables prompt analysis and visualization of critical metrics.

This Real-Time E-commerce Analytics Dashboard facilitates agile decision-making by quickly identifying trends, sales patterns, and growth opportunities. The ability to visualize the best-selling and least-selling products, coupled with insights into customer behavior, enables businesses to tailor strategies for improved customer satisfaction. By offering real-time insights, this dashboard serves as a strategic asset for E-commerce enterprises, enhancing their ability to make informed decisions in the competitive market landscape.

1. INTRODUCTION

The E - commerce (electronic commerce) refers to the purchase and sale of products and services through the electronic network, primarily the internet. These business transactions occur either as business-to-business (B2B), business-to- consumer (B2C), consumer-to-consumer or consumer-to-business. The rapid growth of online shopping has created a saturated market. This can be difficult to navigate unless you keep up with the ever-evolving e Commerce industry.

It contains a large number of data, processes, and tools for customers and sellers, such as smart device shopping, cash on delivery, and online payment encryption. According to a research report, due to the covid-19 pandemic, online sales have increased. Customers are accustomed to submitting reviews or comments after receiving deliveries to share their opinions about the quality of the products and services.

However, for instance of online buying, e-commerce service providers go to great lengths to assist users in selecting trustworthy products. Various businesses, particularly e-commerce, heavily rely on analysis to boost product quality and make the right business decisions in today's competitive business world. E-commerce companies want to know what their customers think about their sellers and products that help them to maintain their online reputation. Shoppers can easily get ideas about which product will be best for them compared to other products using this active dashboard empowers businesses to grasp customer trends, seller dynamics, and product performance in real time.

2. OBJECTIVE OF PROJECT

The objective of this project is to develop a Real-time Ecommerce Analytics Dashboard using tableau visualization. The dashboard should empower businesses to grasp customer trends, seller dynamics, and product performance in real time. The main objectives of the project are as follows:

1. **Data Preprocessing:** The first objective is to preprocess the data by removing missing values, outliers, and noise to improve the quality of the data.
2. **Feature Engineering:** The second objective is to select the most relevant features that can best represent the data and require for the analysis.
3. **Data Streaming:** The third objective is to process data as it is generated, which can be used for a variety of purposes, such as monitoring, analytics at real time. This can be done using Kafka producers and consumers.
4. **MongoDB Configuration:** Establish a MongoDB database instance and create a collection to store the incoming data.
5. **Data Storage:** Develop a script or application that pushes the streaming data into the MongoDB collection in real-time.
6. **JSON Generation:** Write a script to generate a JSON file that acts as a data connection bridge between MongoDB and Tableau. This JSON file should contain relevant connection details and data query instructions.
7. **Tableau Integration:** Open Tableau and import the JSON file as a data connection. Configure Tableau to fetch data from MongoDB using the provided JSON connection.
8. **Visualization:** Design and create visualizations in Tableau using the data from Kafka consumer. Leverage Tableau's features to represent insights effectively.

3. FUNCTIONAL REQUIREMENTS

1) Python 3:

- Python is a high-level programming language that is easy to learn and use.
- Python is an interpreted language, which means that code can be executed on the fly, without theneed for compilation.
- Python is open source and free to use, with a large and active community of developers contributingto its development and maintenance.
- Python has a vast collection of third-party libraries and packages, such as NumPy, Pandas,Matplotlib, Plotly, Seaborn among others, that make it easy to perform data analysis.

2) Kafka:

- Kafka is a distributed event streaming platform designed for high-throughput, fault-tolerant, and real-time data streaming.
- Kafka facilitates the real-time flow of data between various applications or systems in a publish-subscribe model.
- Kafka's architecture allows for horizontal scalability, ensuring high performance and fault tolerance.
- It employs topics to categorize and organize data streams, making it easy to manage different data sources.
- Kafka's message retention capabilities enable data replay and historical analysis, even after it has been consumed.

3) Tableau:

- Tableau is a data visualization and business intelligence software that allows users to connect,analyze, and share data in a visual and interactive way.
- It offers a user-friendly drag-and-drop interface that enables users to create interactive dashboards,reports, and charts without the need for complex coding or programming.
- Tableau supports various data sources, including spreadsheets, databases, cloud services, and big dataplatforms, such as Hadoop and Spark.

4. SYSTEM ARCHITECTURE

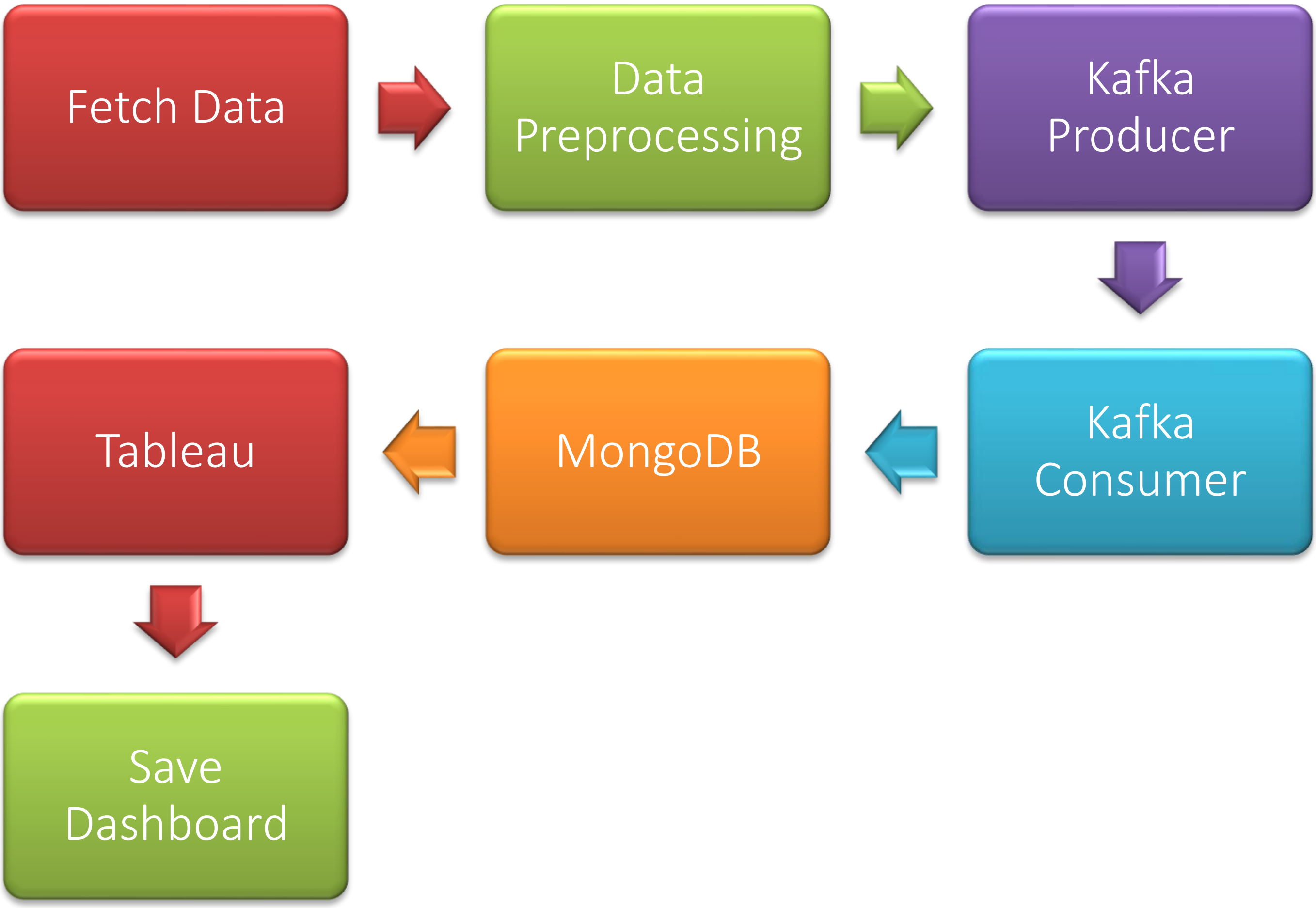

Fig. 4.1 : System Architecture of Real-time Ecommerce Analytics Dashboard

5. Real Time Data Streaming

In our project, we used real time data streaming: -

- Real-time data streaming refers to the continuous and immediate flow of data from various sources to a destination, often with minimal latency.
- Unlike traditional batch processing, where data is collected and processed in predefined intervals, real-time data streaming processes data as it arrives, enabling near-instantaneous analysis and response.
- Real-time streaming minimizes delays between data generation and consumption, enabling swift insights and actions.
- Popular real-time streaming technologies include Apache Kafka, Apache Flink, Amazon Kinesis, and more.
- Based on the aforementioned streaming technologies, we have chosen Apache Kafka due to our familiarity and comfort with the platform.

5.1. Kafka Streaming: -

- **Kafka as the Backbone**:
  - Kafka serves as the backbone of our real-time data processing. It facilitates the seamless flow of data from various sources, enabling us to capture dynamic e-commerce information in real time.
  - Utilizing Kafka's producer API, we can efficiently collect and publish incoming e-commerce data, ensuring minimal latency between data generation and transmission.
- **Kafka Consumers for Immediate Insights:**
  - Kafka consumers play a pivotal role in our project's real-time analytics. They subscribe to relevant Kafka topics, retrieving incoming data streams as soon as they are published.
  - Through the integration of Kafka consumers, we enable immediate processing of data, enabling timely insights into changing e-commerce trends and patterns.
- **JSON Generation for Tableau Integration:**
  - After consuming data from Kafka, our project generates JSON files containing the extracted insights. These JSON files serve as a direct bridge between Kafka consumers and Tableau for visualization.
  - This approach ensures a streamlined and efficient connection between the real-time data stream and Tableau's visualization capabilities, allowing for swift representation of actionable insights.

5.2 Data Storage in MongoDB:

- While the JSON files are directed towards Tableau, we concurrently store the data in MongoDB. This parallel process ensures data persistence for future analysis and reference.
- MongoDB's flexibility and scalability make it an ideal choice for storing real-time e-commerce data, ensuring the availability of historical insights.

5.3 Tableau Integration for Visual Representation:

- By integrating Tableau with the JSON files generated from Kafka consumers, we enable the visualization of e-commerce trends and metrics in real time.
- Tableau's powerful visualization tools allow stakeholders to interact with and interpret data dynamically, fostering informed decision-making.

5.4 Kafka Producer code

```
import time
from json import dumps
from kafka import KafkaProducer
from pyspark.sql import SparkSession

# Define the Kafka topic and bootstrap servers
kafka_topic = "RTEAD"
kafka_bootstrap_servers = "localhost:9092"

def main():
    print("Kafka Producer Application Started ...")

    # Create a Kafka producer object
    kafka_producer_obj = KafkaProducer(bootstrap_servers="localhost:9092",
                                       value_serializer=lambda x: dumps(x).encode('utf-8'))

    # Create a Spark session
    spark = SparkSession.builder.appName("KafkaProducer").getOrCreate()

    # importing csv files
    orders = spark.read.csv(r"D:\ModifiedProject\Data\olist_orders.csv",
header=True)
    customers =
spark.read.csv(r"D:\ModifiedProject\Data\olist_customers_dataset.csv",
header=True)
```

```
order_items =spark.read.csv
(r"D:\ModifiedProject\Data\olist_order_items_dataset.csv", header=True)
    sellers = spark.read.csv(r"D:\ModifiedProject\Data\olist_sellers_dataset.csv",
header=True)
    products = spark.read.csv(r"D:\ModifiedProject\Data\product_translated.csv",
header=True)
    order_payments =
spark.read.csv(r"D:\ModifiedProject\Data\olist_order_payments_dataset.csv",
header=True)

    # Select the desired columns
    orders = orders.select('order_id', 'customer_id', 'order_purchase_timestamp',
'order_status', 'order_delivered_customer_date' , 'order_estimated_delivery_date')
    customers = customers.drop('customer_unique_id')
    order_items = order_items.drop('shipping_limit_date')
    products = products.select('product_id', 'product_category_name_english')
    order_payments = order_payments.select('order_id', 'payment_type')

    # Merge the DataFrames
    Streaming_DataSet = orders.join(customers, on='customer_id', how='outer')
    Streaming_DataSet = Streaming_DataSet.join(order_items, on='order_id',
how='outer')
    Streaming_DataSet = Streaming_DataSet.join(products, on='product_id',
how='outer')
    Streaming_DataSet = Streaming_DataSet.join(order_payments, on='order_id',
how='outer')
    Streaming_DataSet = Streaming_DataSet.join(sellers, on='seller_id', how='outer')
    Streaming_DataSet =
Streaming_DataSet.dropna(subset=['product_category_name_english'])
    Streaming_DataSet = Streaming_DataSet.dropna(subset=['payment_type'])

    # Convert the DataFrame to a list of JSON objects, where each JSON object
represents a row in the DataFrame.
    orders_list = Streaming_DataSet.toJSON().collect()

    # Iterating through the list of JSON objects and sends each object to Kafka.
    for order in orders_list:
        message = order
        print(message)
        kafka_producer_obj.send(kafka_topic, message)
        time.sleep(1)


if __name__ == "__main__":
    main()
```

5.5 Kafka Consumer code

```
import json
from kafka import KafkaConsumer
from pymongo import MongoClient

# Define the Kafka topic and bootstrap servers that the consumer will connect to.
kafka_topic = "RTEAD"
kafka_bootstrap_servers = "localhost:9092"

# Set up MongoDB connection
client = MongoClient('mongodb://localhost:27017/')
db = client['project']
collection = db['RTEAD']

def main():
    print("Kafka Consumer Application Started ...")

    kafka_consumer_obj = KafkaConsumer(kafka_topic,
bootstrap_servers=kafka_bootstrap_servers)

    # Create a JSON file to store the messages
    json_file = open(r"D:\ModifiedProject\Output\RETADModified2.json", "a")
    i = 0
    for message in kafka_consumer_obj:
        kafka_data = json.loads(message.value)

        # Append the message to the JSON file
        json_file.write(kafka_data + "\n")
        # Creating dictionary of from JSON String
        obj_dict = eval(kafka_data)

        # Set the _id field value to i+1
        obj_dict["_id"] = i + 1
        print(obj_dict)

        collection.update_one(
            {'_id': obj_dict['_id']},
            {'$set': obj_dict},
            upsert=True  # Insert if not exists, otherwise update
        )

        i += 1

    # Close the JSON file
    json_file.close()


if __name__ == "__main__":
    main()
```

6. DATA VISUALIZATION AND REPRESENTATION

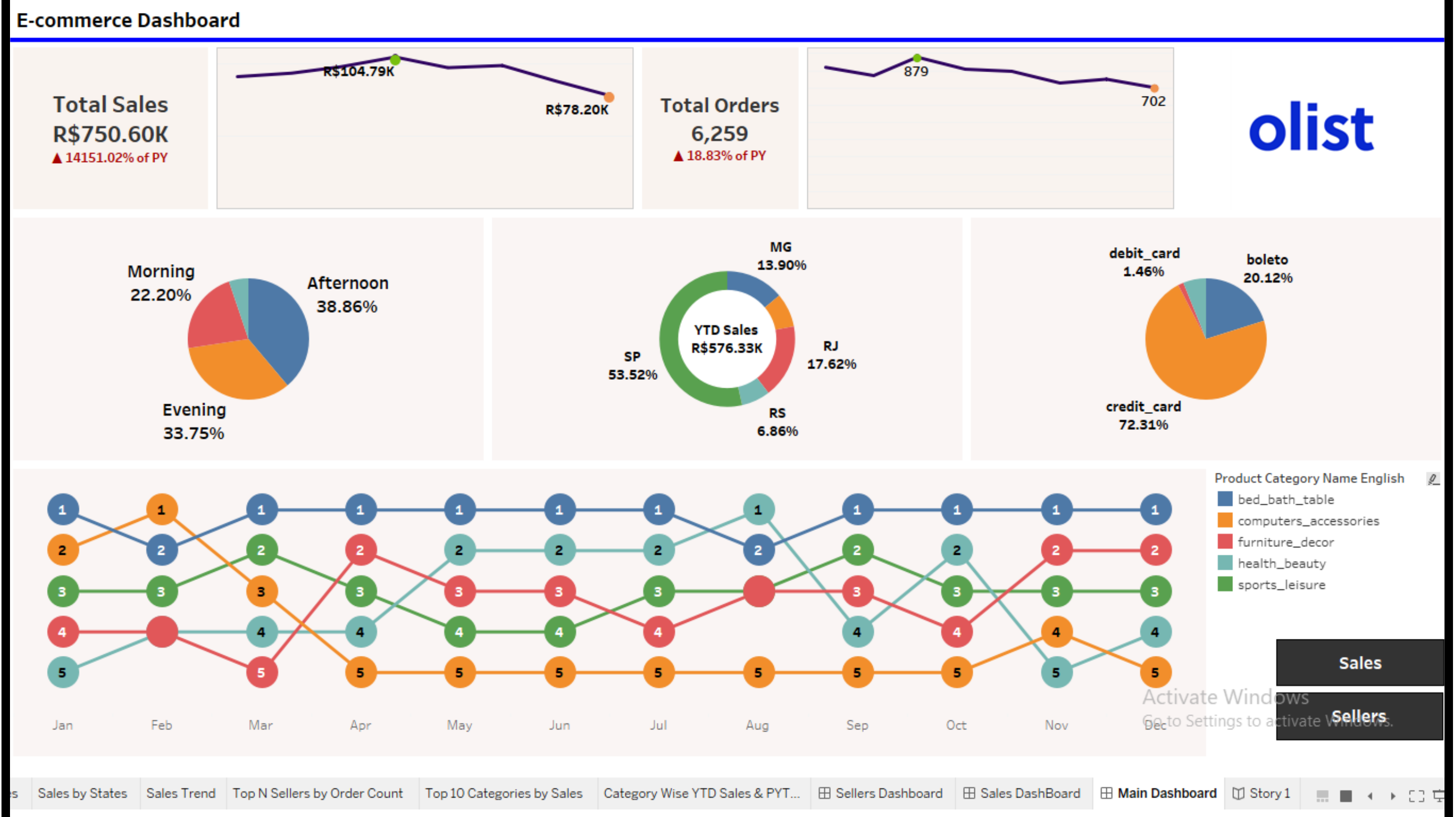

Fig. 6.1 : Main Dashboard

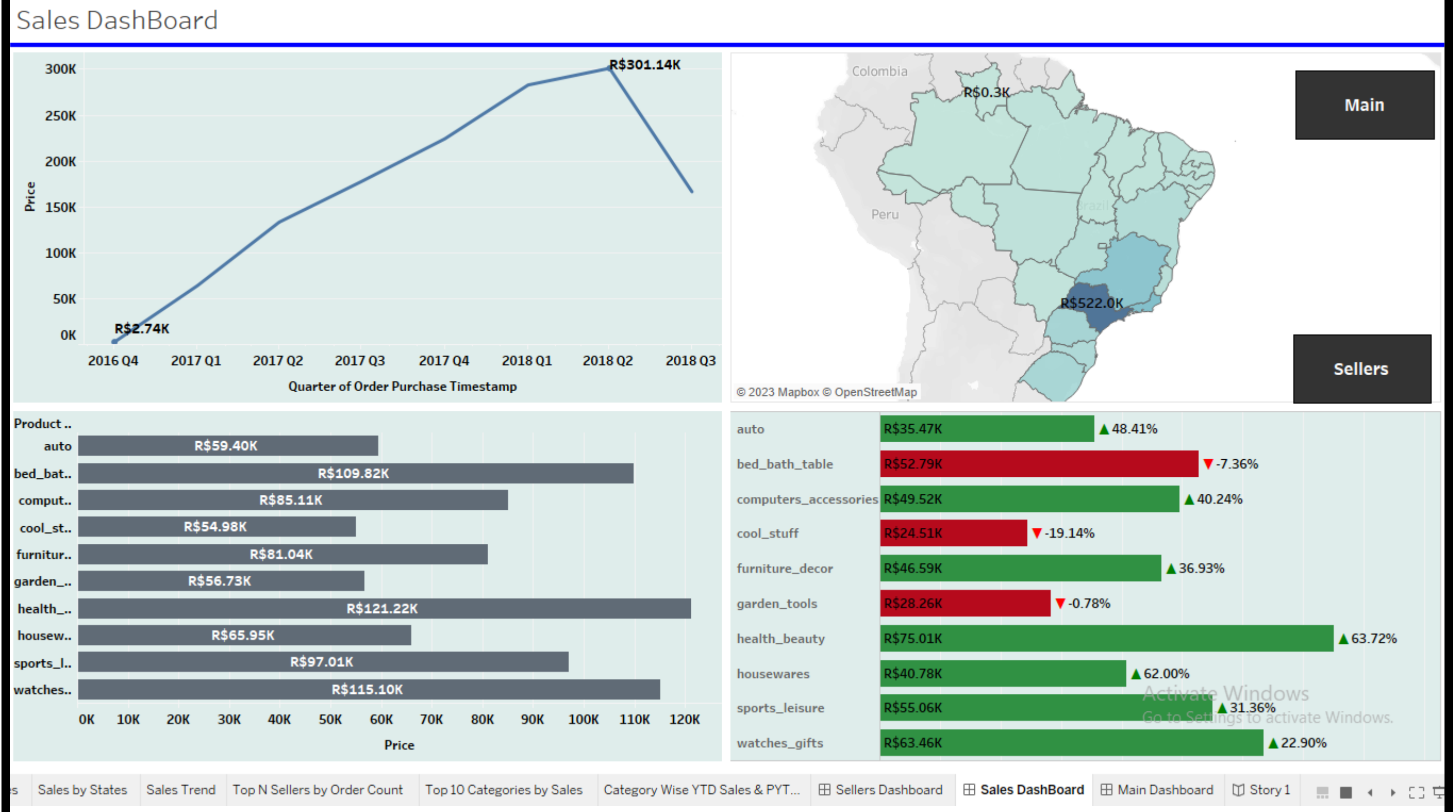
Sales DashBoard
300K
250K
200K
150K
100K
50K
0K
Price
R$301.14K
R$2.74K
2016 Q4
2017 Q1
2017 Q2
2017 Q3
2017 Q4
2018 Q1
2018 Q2
2018 Q3
Quarter of Order Purchase Timestamp
Colombia
Peru
R$0.3K
R$522.0K
© 2023 Mapbox © OpenStreetMap
Main
Sellers
Product ..
auto R$59.40K
bed_bat.. R$109.82K
comput.. R$85.11K
cool_st.. R$54.98K
furnitur.. R$81.04K
garden_.. R$56.73K
health_.. R$121.22K
housew.. R$65.95K
sports_l.. R$97.01K
watches.. R$115.10K
0K 10K 20K 30K 40K 50K 60K 70K 80K 90K 100K 110K 120K
Price
auto R$35.47K ▲48.41%
bed_bath_table R$52.79K ▼-7.36%
computers_accessories R$49.52K ▲40.24%
cool_stuff R$24.51K ▼-19.14%
furniture_decor R$46.59K ▲36.93%
garden_tools R$28.26K ▼-0.78%
health_beauty R$75.01K ▲63.72%
housewares R$40.78K ▲62.00%
sports_leisure R$55.06K ▲31.36%
watches_gifts R$63.46K ▲22.90%
Activate Windows
Go to Settings to activate Windows.
Sales by States
Sales Trend
Top N Sellers by Order Count
Top 10 Categories by Sales
Category Wise YTD Sales & PYT...
Sellers Dashboard
Sales DashBoard
Main Dashboard
Story 1

Fig. 6.2 : Sales Dashboard

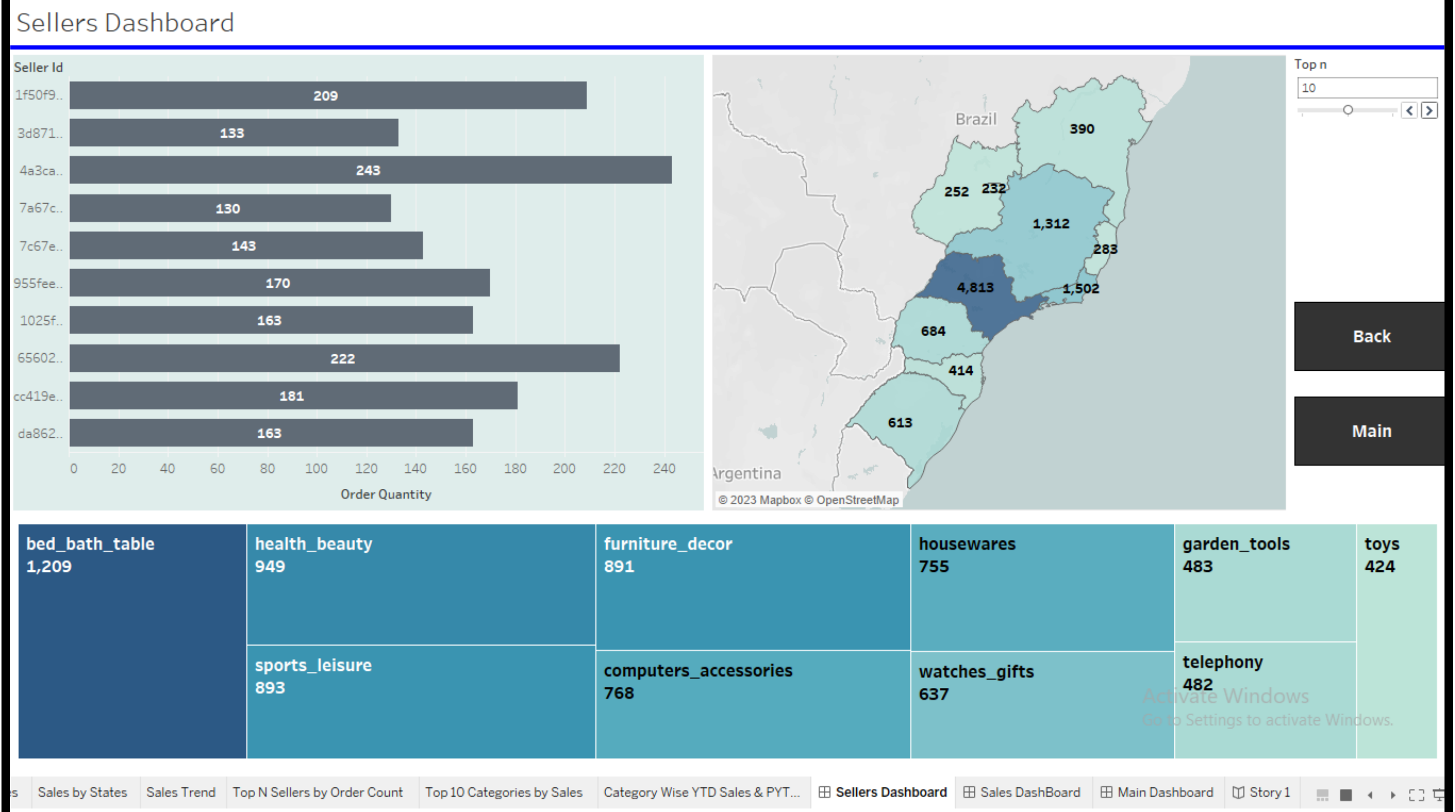

Fig. 6.3 : Sellers Dashboard

7. CONCLUSION AND FUTURE SCOPE

- In conclusion, Our project has successfully demonstrated the potential of real-time analytics in the e-commerce sector. By leveraging Apache Kafka's streaming capabilities, we've enabled stakeholders to access and interpret up-to-the-minute insights, translating data into informed decisions.

- Tableau visualization yielded an interactive and intuitive representation of e-commerce metrics. This visual interface empowers users to explore trends and draw actionable insights with ease.

- Our Real-Time Analytics Dashboard project stands as a testament to the transformative potential of real-time data processing and visualization. By embracing this approach, businesses can navigate the dynamic e-commerce landscape with agility, extract actionable insights, and proactively steer their strategies towards growth and success.

- In future scope we can Enhance the dashboard by incorporating real-time alert mechanisms that trigger notifications for critical events, anomalies, or significant changes in key performance indicators. This feature would enable stakeholders to respond promptly to emerging situations.

8. FUTURE ENHANCEMENT

- The Kafka's performance may present limitations for our project's scalability and real-time processing demands:
- Latency in Complex Processing: Kafka's processing model, while efficient for simple operations, can introduce latency in complex data transformations, impacting real-time analytics.
- Future Enhancement: Spark Integration for Enhanced Performance
- Integrating Apache Spark can address Kafka's performance challenges and provide substantial benefits:
- By considering the integration of Apache Spark, we can overcome Kafka's performance limitations and unlock a higher level of scalability, speed, and complexity handling within our Real-Time E-Commerce Analytics Dashboard.

References

1. https://www.kaggle.com/datasets/olistbr/brazilian-ecommerce

2. Exploratory Data Analysis and Visualization of Olist (Brazilian E-commerce) Public Dataset | by Rona Fauzan Noer | Medium

3. “Big Data Analytics for E-commerce: A Review" by J. Lu and J. Zhang.

4. Exploratory Data Analysis (EDA) on Brazilian E-Commerce Olist | by Yash Bhairao | Jovian