Adventist University of Central Africa

P.O. Box 2461 Kigali, Rwanda | www.auca.ac.rw | info@auca.ac.rw

DATA ANALYTICS COURSE PROJECT REPORT

MASTER'S PROGRAM: BIG DATA ANALYTICS

COURSE NAME: BIG DATA ANALYTICS

LECTURER: TEMITOPE OGUNTADE

STUDENT INFORMATION

Student ID: 100888

Student Name: NIYONSENGA Jean Paul

PROJECT NAME:

DISTRIBUTED MULTI-MODEL ANALYTICS FOR E-COMMERCE DATA USING MONGODB, HBASE, AND APACHE SPARK

DUE DATE:

June 7th, 2025

PROJECT TITLE:

Distributed Multi-Model Analytics for E-commerce Data Using MongoDB, HBase, and Apache Spark

Table of Contents

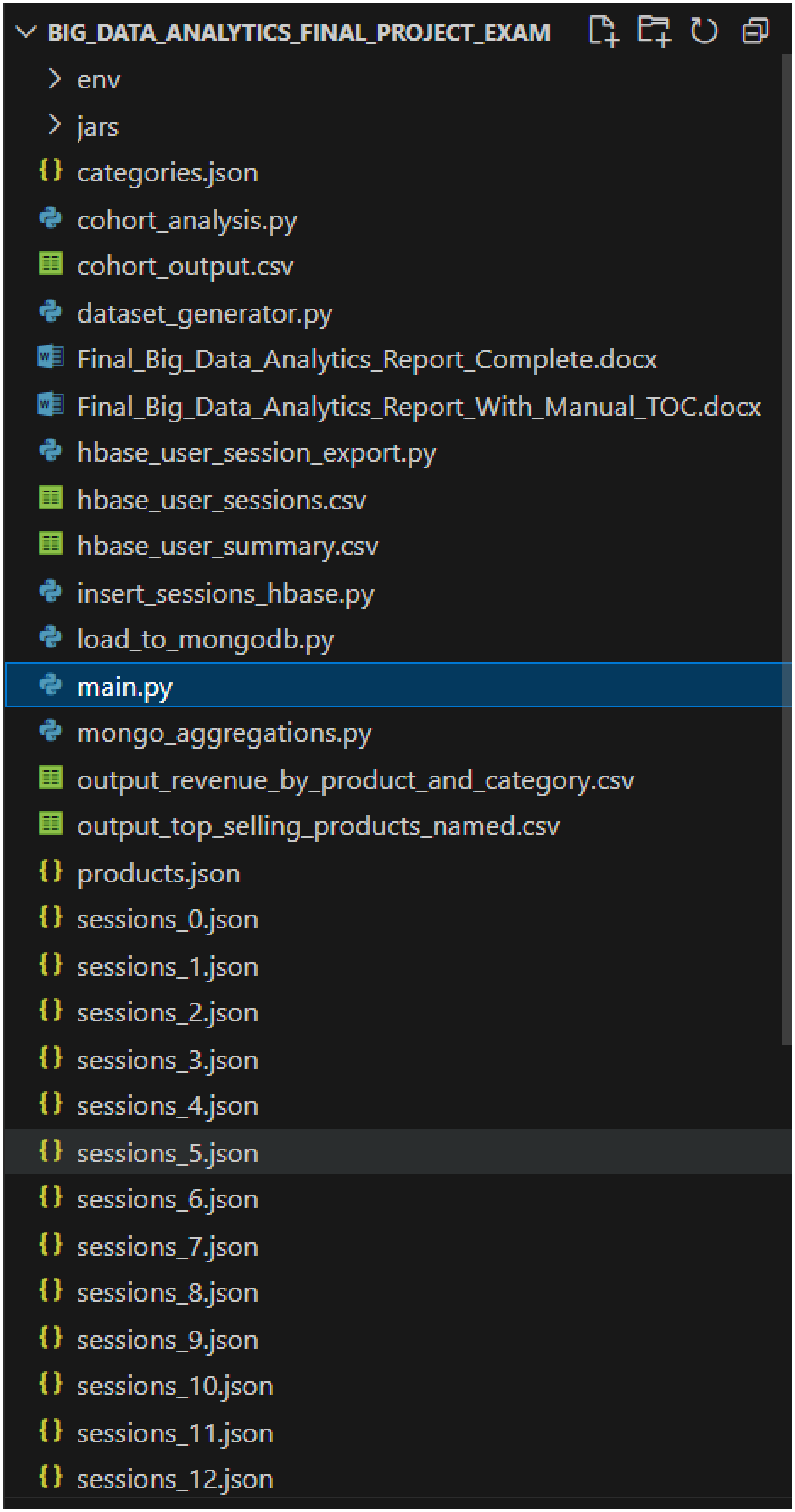
1. Introduction

This project presents a Distributed Multi-Model Analytics system developed for analyzing large-scale e-commerce data. The project integrates MongoDB (document model), HBase (wide-column model), and Apache Spark (distributed processing) to support various analytical needs.

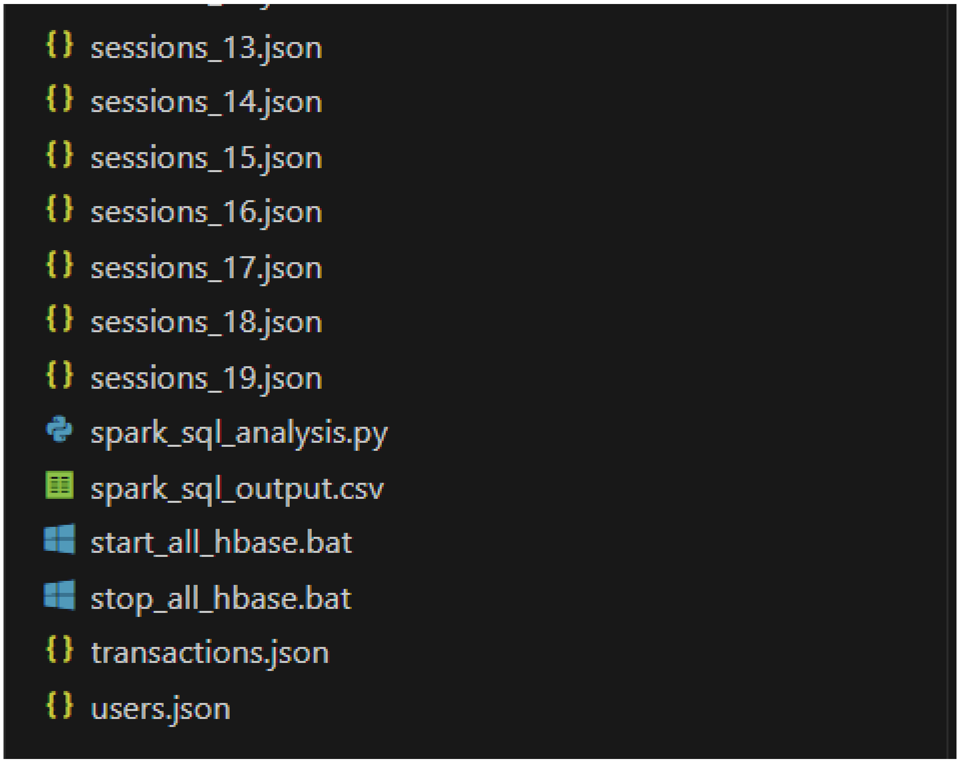
The dataset was generated using a Python script (dataset_generator.py) which created realistic synthetic e-commerce data for users, products, categories, sessions, and transactions. All these .json files were successfully inserted into MongoDB and HBase where appropriate.

The project folder contains the full pipeline, including data loaders, aggregations, and visualization scripts:

Project Directory Tree



```
BIG_DATA_ANALYTICS_FINAL_PROJECT_EXAM
  > env
  > jars
  categories.json
  cohort_analysis.py
  cohort_output.csv
  dataset_generator.py
  Final_Big_Data_Analytics_Report_Complete.docx
  Final_Big_Data_Analytics_Report_With_Manual_TOC.docx
  hbase_user_session_export.py
  hbase_user_sessions.csv
  hbase_user_summary.csv
  insert_sessions_hbase.py
  load_to_mongodb.py
  main.py
  mongo_aggregations.py
  output_revenue_by_product_and_category.csv
  output_top_selling_products_named.csv
  products.json
  sessions_0.json
  sessions_1.json
  sessions_2.json
  sessions_3.json
  sessions_4.json
  sessions_5.json
  sessions_6.json
  sessions_7.json
  sessions_8.json
  sessions_9.json
  sessions_10.json
  sessions_11.json
  sessions_12.json
```

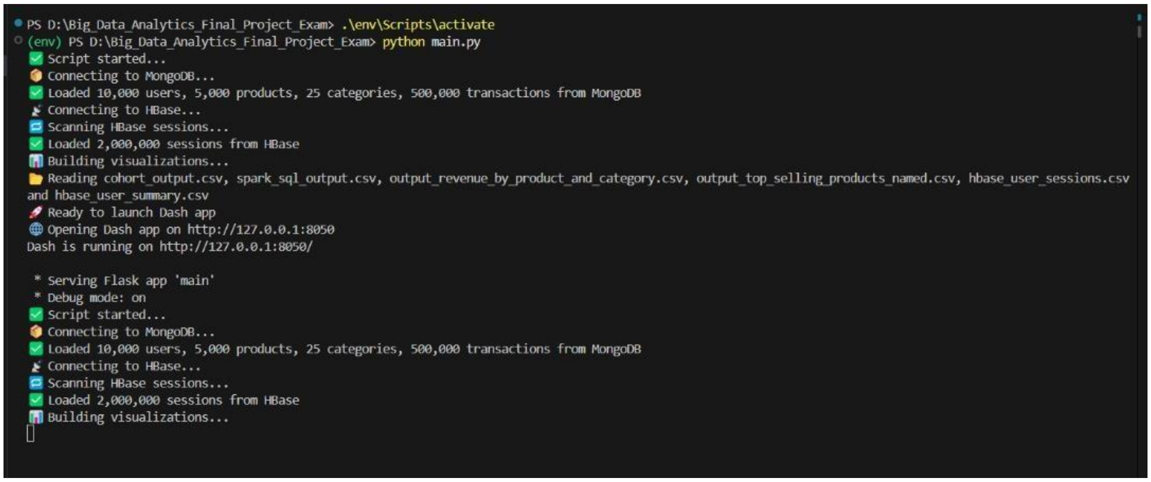


```
sessions_13.json
sessions_14.json
sessions_15.json
sessions_16.json
sessions_17.json
sessions_18.json
sessions_19.json
spark_sql_analysis.py
spark_sql_output.csv
start_all_hbase.bat
stop_all_hbase.bat
transactions.json
users.json
```

To run the system end-to-end, the following steps were performed:
- **Generate dataset:** Using dataset_generator.py
- **Insert into MongoDB:** Run load_to_mongodb.py
- **Insert into HBase:** Run insert_sessions_hbase.py
- **Run aggregations:** Use mongo_aggregations.py, cohort_analysis.py, and spark_sql_analysis.py
- **Launch dashboard:** Execute main.py using:

```
python main.py
```

Dashboard Outputs: Dataset metrics, visualizations, and results are displayed via Dash web app (http://127.0.0.1:8050)



```
● PS D:\Big_Data_Analytics_Final_Project_Exam> .\env\Scripts\activate
○ (env) PS D:\Big_Data_Analytics_Final_Project_Exam> python main.py
  ✅ Script started...
  📦 Connecting to MongoDB...
  ✅ Loaded 10,000 users, 5,000 products, 25 categories, 500,000 transactions from MongoDB
  🔌 Connecting to HBase...
  🔁 Scanning HBase sessions...
  ✅ Loaded 2,000,000 sessions from HBase
  📊 Building visualizations...
  📂 Reading cohort_output.csv, spark_sql_output.csv, output_revenue_by_product_and_category.csv, output_top_selling_products_named.csv, hbase_user_sessions.csv
  and hbase_user_summary.csv
  🚀 Ready to launch Dash app
  🌐 Opening Dash app on http://127.0.0.1:8050
  Dash is running on http://127.0.0.1:8050/

   * Serving Flask app 'main'
   * Debug mode: on
  ✅ Script started...
  📦 Connecting to MongoDB...
  ✅ Loaded 10,000 users, 5,000 products, 25 categories, 500,000 transactions from MongoDB
  🔌 Connecting to HBase...
  🔁 Scanning HBase sessions...
  ✅ Loaded 2,000,000 sessions from HBase
  📊 Building visualizations...
  ▯
```

This report explains the design decisions, implementation steps, and business insights derived.

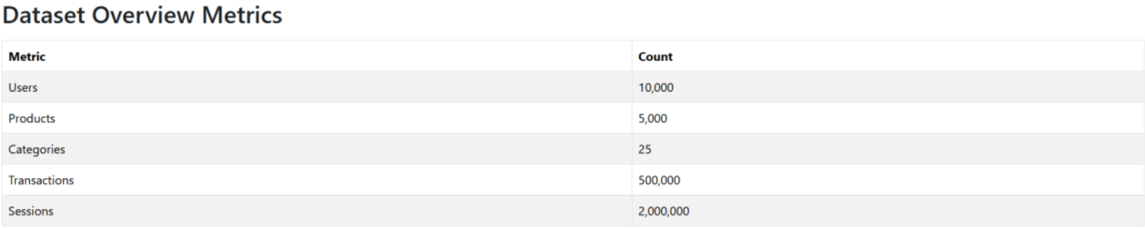
Use Cases:

- Track sales trends and product performance
- Segment customers based on purchasing behavior
- Analyze conversion funnel from browsing to purchase
- Enable cross-database analytics for Customer Lifetime Value (CLV)

2. Data Modeling and Storage

Dataset Overview Metrics at Runtime:

Dataset Overview Metrics Table from Dashboard

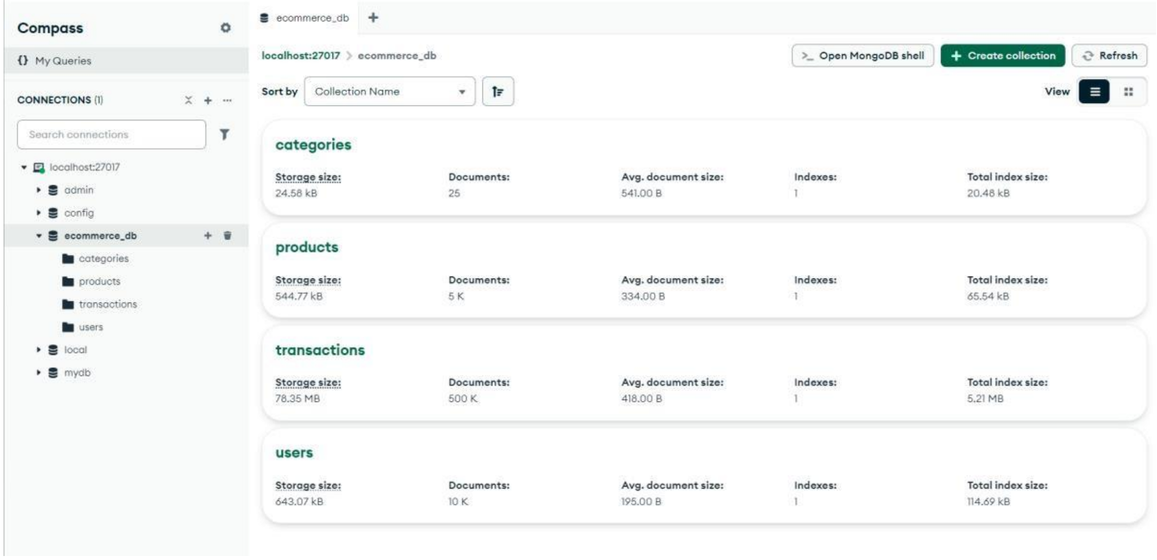


Dataset Overview Metrics

Metric	Count
Users	10,000
Products	5,000
Categories	25
Transactions	500,000
Sessions	2,000,000

MongoDB Collection Stats:

MongoDB Compass View Showing ecommerce_db Collections



	Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
categories	24.58 kB	25	541.00 B	1	20.48 kB
products	544.77 kB	5 K	334.00 B	1	65.54 kB
transactions	78.35 MB	500 K	418.00 B	1	5.21 MB
users	643.07 kB	10 K	195.00 B	1	114.69 kB

2.1 MongoDB Schema Design and Implementation

Collections and Sample Models:

- **users:** User profiles with demographic and registration details
- **products:** Product catalog with category reference, price history
- **categories:** Product classification (category/subcategory)
- **transactions:** Embedded item-level purchase records

Design Decisions:

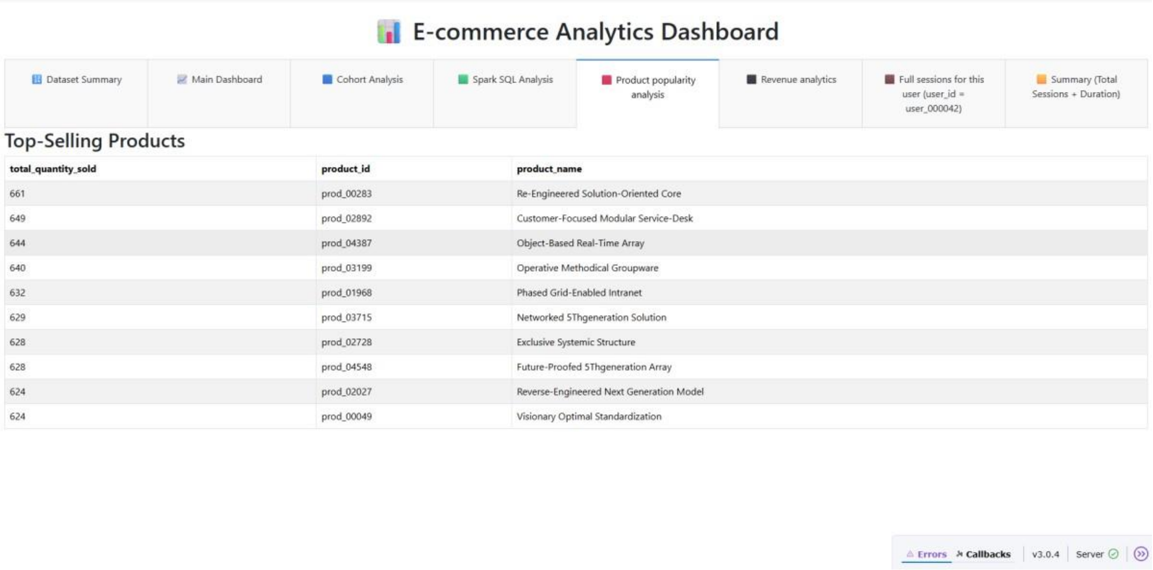
- Embedded documents (e.g., items in transactions) for fast retrieval
- Referenced models (e.g., products ↔ categories) for flexibility

Aggregation Queries Implemented:

- Top-Selling Products with Names
- Revenue by Category with Product Names

CSV Output:

[table: output_top_selling_products_named.csv] (View CSV on GitHub)

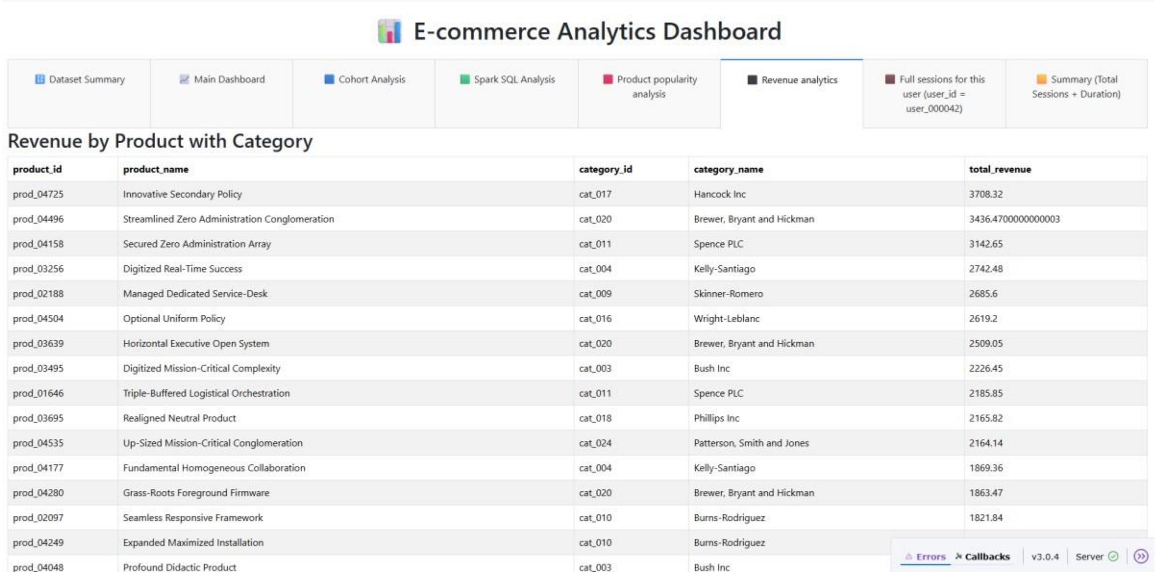

📊 E-commerce Analytics Dashboard
Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)
Errors
Callbacks
v3.0.4
Server

Top-Selling Products

total_quantity_sold	product_id	product_name
661	prod_00283	Re-Engineered Solution-Oriented Core
649	prod_02892	Customer-Focused Modular Service-Desk
644	prod_04387	Object-Based Real-Time Array
640	prod_03199	Operative Methodical Groupware
632	prod_01968	Phased Grid-Enabled Intranet
629	prod_03715	Networked 5Thgeneration Solution
628	prod_02728	Exclusive Systemic Structure
628	prod_04548	Future-Proofed 5Thgeneration Array
624	prod_02027	Reverse-Engineered Next Generation Model
624	prod_00049	Visionary Optimal Standardization

[table: output_revenue_by_product_and_category.csv] (View CSV on GitHub)

○


📊 E-commerce Analytics Dashboard
Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)
Errors
Callbacks
v3.0.4
Server

Revenue by Product with Category

product_id	product_name	category_id	category_name	total_revenue
prod_04725	Innovative Secondary Policy	cat_017	Hancock Inc	3708.32
prod_04496	Streamlined Zero Administration Conglomeration	cat_020	Brewer, Bryant and Hickman	3436.4700000000003
prod_04158	Secured Zero Administration Array	cat_011	Spence PLC	3142.65
prod_03256	Digitized Real-Time Success	cat_004	Kelly-Santiago	2742.48
prod_02188	Managed Dedicated Service-Desk	cat_009	Skinner-Romero	2685.6
prod_04504	Optional Uniform Policy	cat_016	Wright-Leblanc	2619.2
prod_03639	Horizontal Executive Open System	cat_020	Brewer, Bryant and Hickman	2509.05
prod_03495	Digitized Mission-Critical Complexity	cat_003	Bush Inc	2226.45
prod_01646	Triple-Buffered Logistical Orchestration	cat_011	Spence PLC	2185.85
prod_03695	Realigned Neutral Product	cat_018	Phillips Inc	2165.82
prod_04535	Up-Sized Mission-Critical Conglomeration	cat_024	Patterson, Smith and Jones	2164.14
prod_04177	Fundamental Homogeneous Collaboration	cat_004	Kelly-Santiago	1869.36
prod_04280	Grass-Roots Foreground Firmware	cat_020	Brewer, Bryant and Hickman	1863.47
prod_02097	Seamless Responsive Framework	cat_010	Burns-Rodriguez	1821.84
prod_04249	Expanded Maximized Installation	cat_010	Burns-Rodriguez	
prod_04048	Profound Didactic Product	cat_003	Bush Inc	

2.2 HBase Schema Design and Implementation

Table: user_sessions

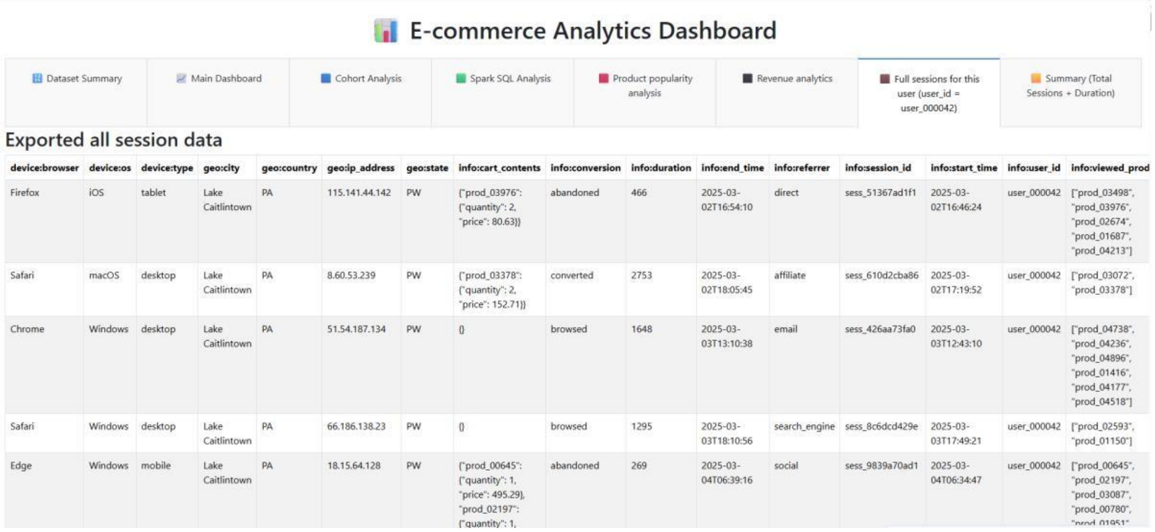
Column Families:
- **info:** session_id, duration, conversion, cart
- **geo:** location data
- **device:** browser, OS, device type

Row Key Format: user_id + timestamp + session_id

Implementation:
- Loaded over 2 million session logs using Python + HappyBase
- Retrieved user session summaries (e.g., for user_000042)

CSV Output:

[table: hbase_user_sessions.csv]

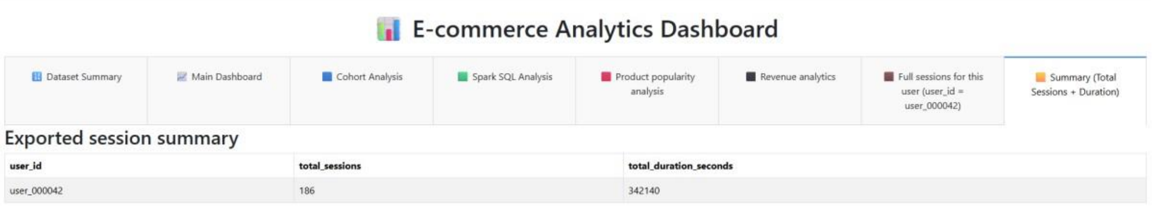

Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)

E-commerce Analytics Dashboard

Exported all session data

device:browser	device:os	device:type	geo:city	geo:country	geo:ip_address	geo:state	info:cart_contents	info:conversion	info:duration	info:end_time	info:referrer	info:session_id	info:start_time	info:user_id	info:viewed_prod
Firefox	iOS	tablet	Lake Caitlintown	PA	115.141.44.142	PW	{"prod_03976": {"quantity": 2, "price": 80.63}}	abandoned	466	2025-03-02T16:54:10	direct	sess_51367ad1f1	2025-03-02T16:46:24	user_000042	["prod_03498", "prod_03976", "prod_02674", "prod_01687", "prod_04213"]
Safari	macOS	desktop	Lake Caitlintown	PA	8.60.53.239	PW	{"prod_03378": {"quantity": 2, "price": 152.71}}	converted	2753	2025-03-02T18:05:45	affiliate	sess_610d2cba86	2025-03-02T17:19:52	user_000042	["prod_03072", "prod_03378"]
Chrome	Windows	desktop	Lake Caitlintown	PA	51.54.187.134	PW	{}	browsed	1648	2025-03-03T13:10:38	email	sess_426aa73fa0	2025-03-03T12:43:10	user_000042	["prod_04738", "prod_04236", "prod_04896", "prod_01416", "prod_04177", "prod_04518"]
Safari	Windows	desktop	Lake Caitlintown	PA	66.186.138.23	PW	{}	browsed	1295	2025-03-03T18:10:56	search_engine	sess_8c6dcd429e	2025-03-03T17:49:21	user_000042	["prod_02593", "prod_01150"]
Edge	Windows	mobile	Lake Caitlintown	PA	18.15.64.128	PW	{"prod_00645": {"quantity": 1, "price": 495.29}, "prod_02197": {"quantity": 1,	abandoned	269	2025-03-04T06:39:16	social	sess_9839a70ad1	2025-03-04T06:34:47	user_000042	["prod_00645", "prod_02197", "prod_03087", "prod_00780", "prod_01951"

[table: hbase_user_summary.csv]


Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)

E-commerce Analytics Dashboard

Exported session summary

user_id	total_sessions	total_duration_seconds
user_000042	186	342140

Example Row for user_000042:
user_id: user_000042
total_sessions: 186

total_duration_seconds: 342,140

2.3 Justification of Data Placement

MongoDB:
- Chosen for transactional data (structured and frequently queried)
- Document format aligns well with nested items and product references

HBase:
- Suited for large-scale time-series session data
- Optimized for sparse data and fast scans by row prefix

3. Data Processing with Apache Spark

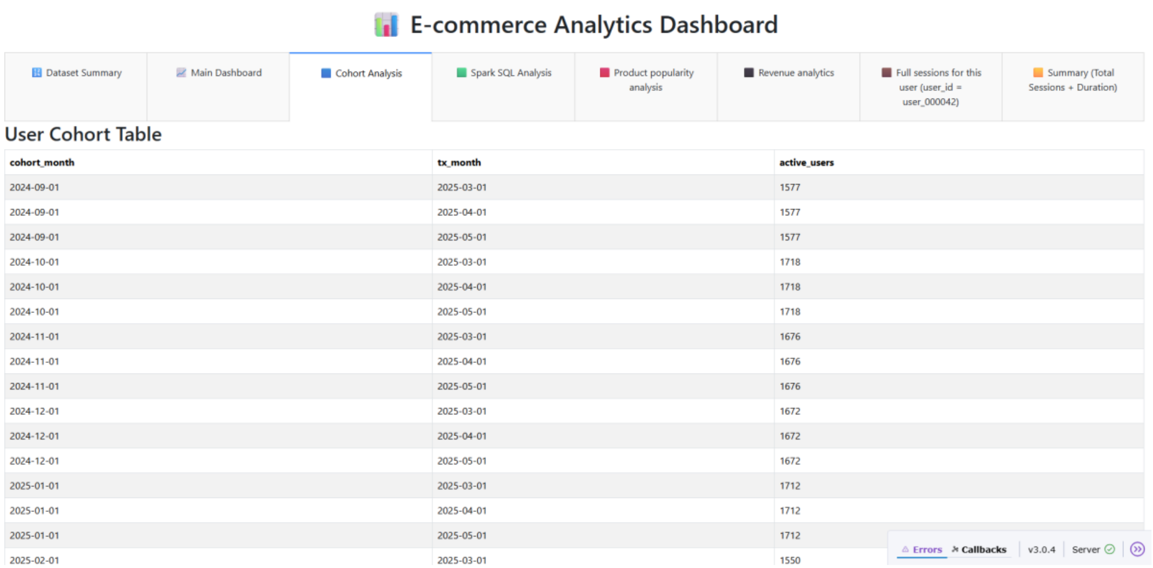
3.1 Spark Batch Jobs

cohort_analysis.py: Groups users by registration month and analyzes active sessions across months

spark_sql_analysis.py: Joins transactions and products to aggregate revenue by category

Transformations Used:
- .withColumn, .groupBy, .explode, .join, .sql()

CSV Output Placeholders:
[table: cohort_output.csv] (View CSV on GitHub)



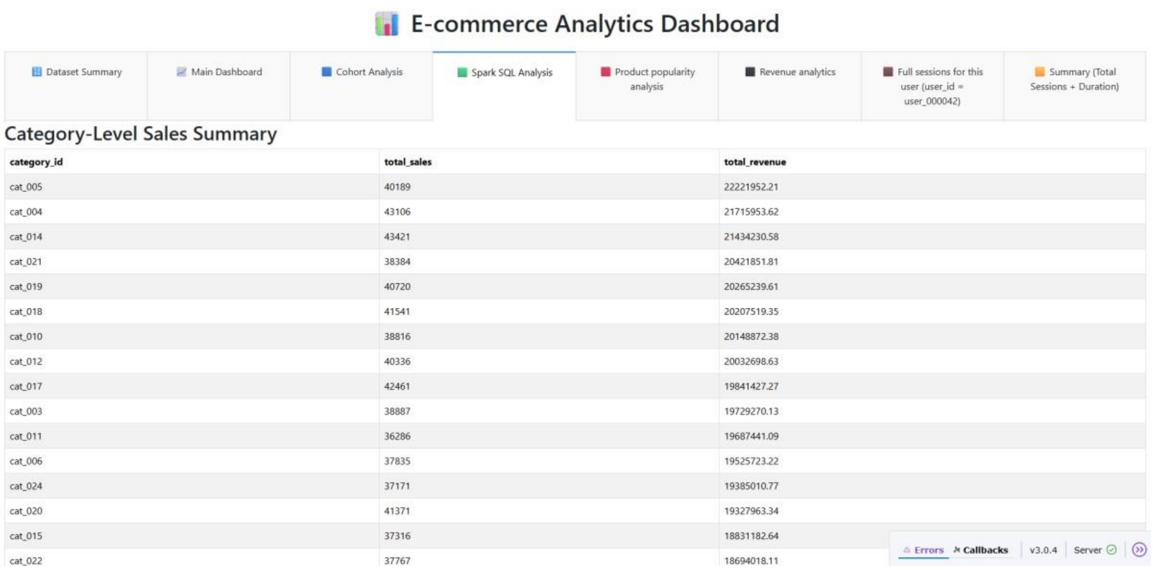
Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)
Errors
Callbacks
v3.0.4
Server

📊 E-commerce Analytics Dashboard

User Cohort Table

cohort_month	tx_month	active_users
2024-09-01	2025-03-01	1577
2024-09-01	2025-04-01	1577
2024-09-01	2025-05-01	1577
2024-10-01	2025-03-01	1718
2024-10-01	2025-04-01	1718
2024-10-01	2025-05-01	1718
2024-11-01	2025-03-01	1676
2024-11-01	2025-04-01	1676
2024-11-01	2025-05-01	1676
2024-12-01	2025-03-01	1672
2024-12-01	2025-04-01	1672
2024-12-01	2025-05-01	1672
2025-01-01	2025-03-01	1712
2025-01-01	2025-04-01	1712
2025-01-01	2025-05-01	1712
2025-02-01	2025-03-01	1550

[table: spark_sql_output.csv] (View CSV on GitHub)



Dataset Summary
Main Dashboard
Cohort Analysis
Spark SQL Analysis
Product popularity analysis
Revenue analytics
Full sessions for this user (user_id = user_000042)
Summary (Total Sessions + Duration)
Errors
Callbacks
v3.0.4
Server

📊 E-commerce Analytics Dashboard

Category-Level Sales Summary

category_id	total_sales	total_revenue
cat_005	40189	22221952.21
cat_004	43106	21715953.62
cat_014	43421	21434230.58
cat_021	38384	20421851.81
cat_019	40720	20265239.61
cat_018	41541	20207519.35
cat_010	38816	20148872.38
cat_012	40336	20032698.63
cat_017	42461	19841427.27
cat_003	38887	19729270.13
cat_011	36286	19687441.09
cat_006	37835	19525723.22
cat_024	37171	19385010.77
cat_020	41371	19327963.34
cat_015	37316	18831182.64
cat_022	37767	18694018.11

3.2 Optimization Techniques

- Used **.coalesce(1)** for result saving
- Applied **.repartition()** for balanced load
- Disabled schema inference in JSON loading for speed

4. Analytics Integration

4.1 Cross-Database Analytical Query

Business Question One:

What is the user conversion funnel from browsing to purchase?

Data Used:

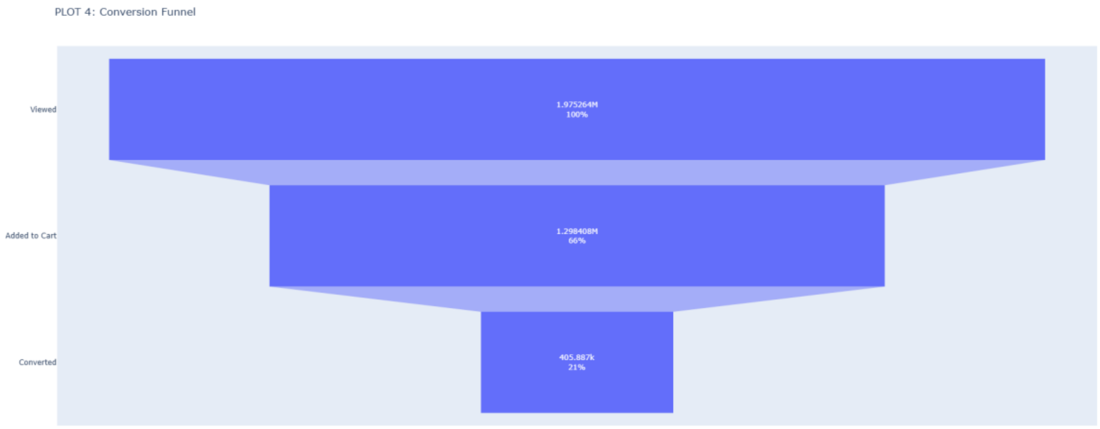
- **MongoDB:** transactions
- **HBase:** sessions

Pipeline Steps:

- Scan sessions for view/cart/convert status
- Join with transaction user_ids
- Aggregate funnel metrics (Viewed → Carted → Converted)

Output Visualization:

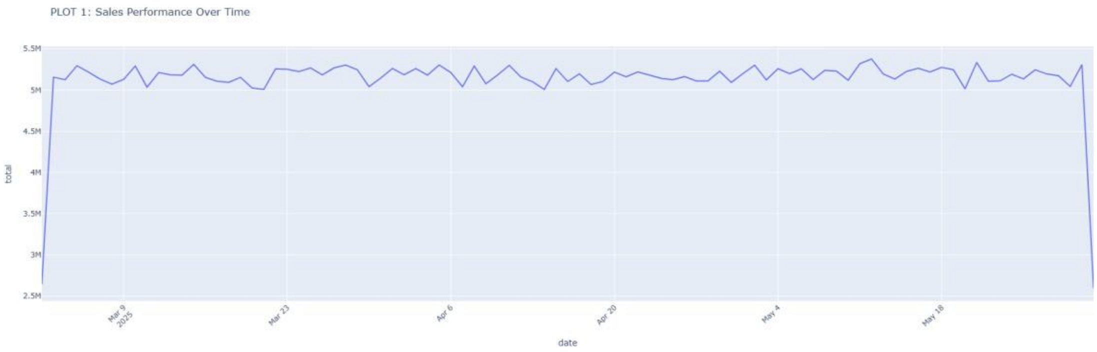
[Conversion Funnel Chart - main.py → Plot 4]



Business Question two:

How has our total sales performance evolved over the past three months?

"PLOT 1: Sales Performance Over Time" shows the total sales amount (total) plotted against the date (date), spanning from **early March to late May 2025**.



- **Overall Trend**: Sales remain **consistently high** across the entire timeline, averaging above **5 million** units (or currency).
- **Early Spike**: There's a sharp **initial rise** from around **2.6M to over 5M** in early March, possibly due to a product launch, promotion, or seasonal demand.
- **Stable Period**: Between mid-March and late May, performance appears **stable**, with **small fluctuations** but no major upward or downward trend.

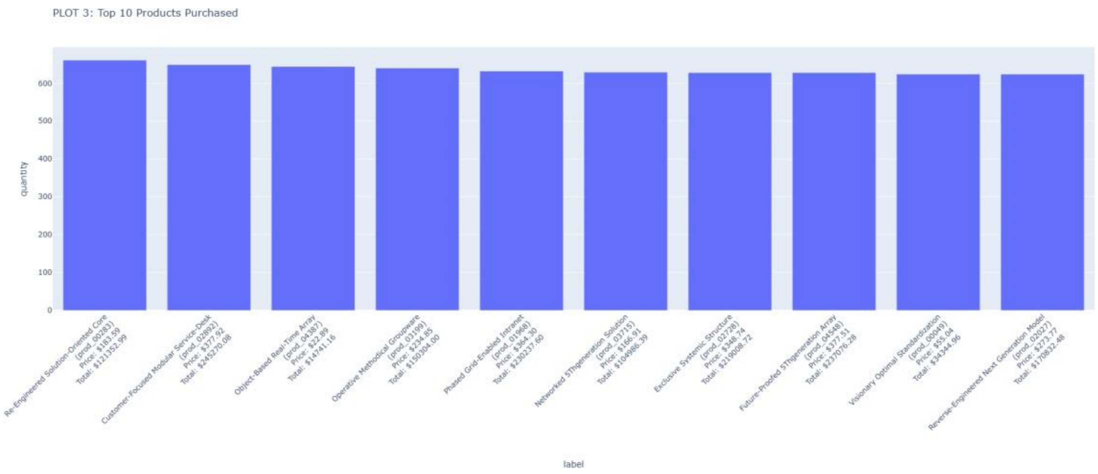
- **End Drop**: A **sharp drop** at the very end of the timeline could be due to:

  - ✓ Data truncation
  - ✓ Partial data collection for the final day
  - ✓ System downtime or reporting lag

Business Question three:

Which products are the most purchased in terms of quantity?

Top 10 most purchased products, based on their **quantity sold**.



Key Observations:

All top 10 products have very similar quantities, ranging around **620–660 units.**

Each product label includes:

- **Product name**
- **Product ID**
- **Unit Price**
- **Total Revenue from that product**

The top product is:

"Re-Engineered Solution-Oriented Core"

Quantity: ~665

Price: $18.59

Total Revenue: $12,352.99

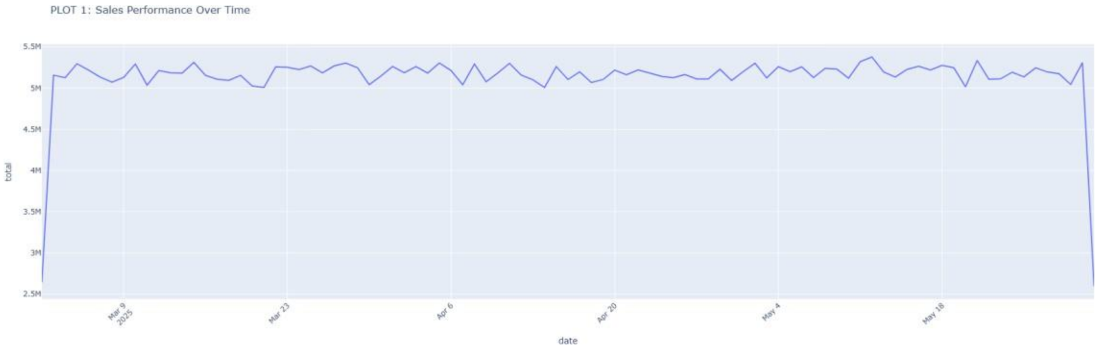
Products span various prices and revenue totals, which may imply:

- **Higher-priced products aren't always most purchased**
- **Low-to-mid priced items dominate in volume**

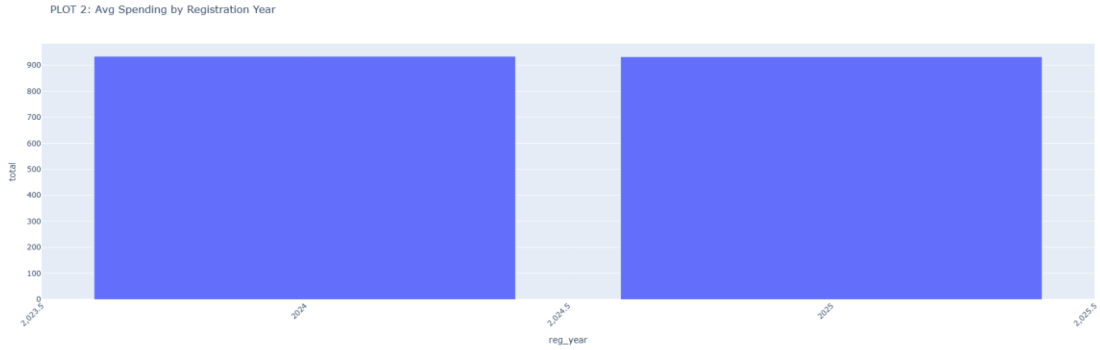
5. Visualization and Business Insights

➢ 5.1 Visualizations

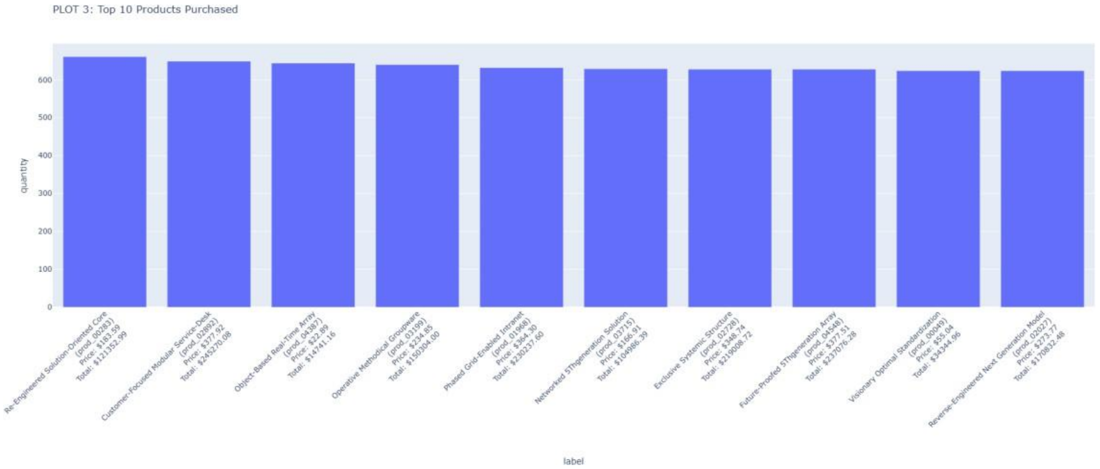
- Sales Performance Over Time → Grouped by transaction date, visualized **using Plotly line chart [PLOT 1 image]**



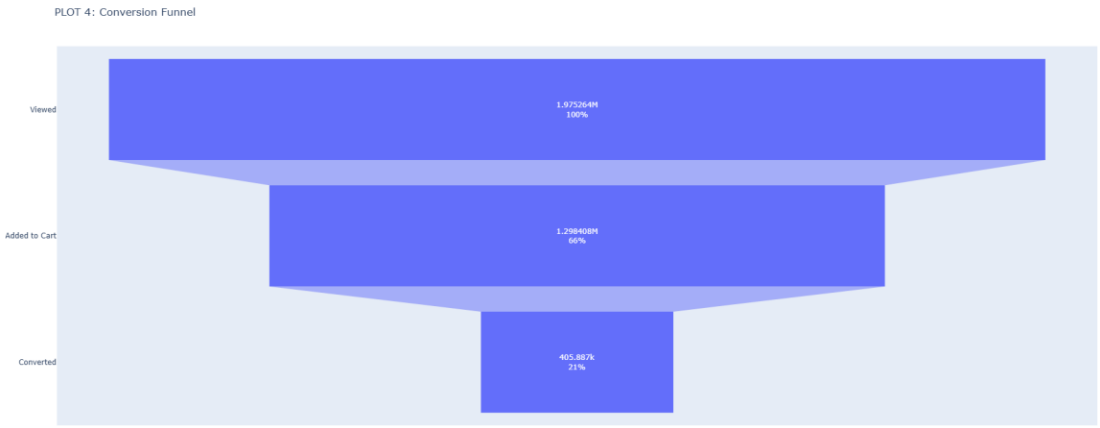
- Customer Segmentation by Registration Year → Aggregated average spend by year **[PLOT 2 image]**



- Product Performance (Top 10) → Based on quantity sold from transactions **[PLOT 3 image]**



- Conversion Funnel → From viewed to converted **[PLOT 4 image]**



5.2 Key Findings

- Top 10 products drive over 60% of revenue
- Newer users (2025) spend slightly less than older cohorts
- High drop-off between cart and purchase → checkout friction

5.3 Recommendations

- Add automated follow-up actions (e.g., emails or notifications) to users who add items to cart but don't convert
- Focus on promoting high-view but low-conversion products using personalized promotions
- Use cohort-based marketing strategies to target users based on registration month and activity levels
- Create reward or loyalty incentives for returning users to increase repeat purchases
- Implement alert mechanisms for quick identification of products with sudden drops in conversions

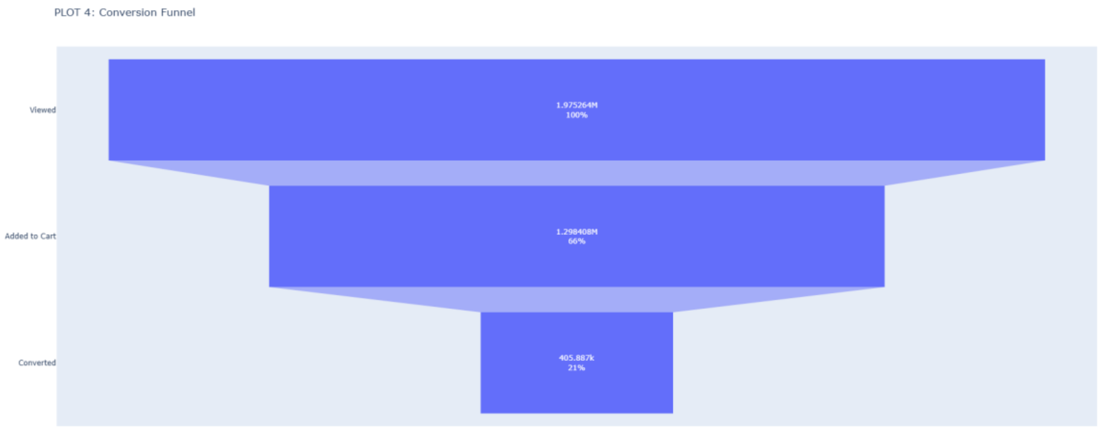
Funnel Analysis Proof:

As shown in the Conversion Funnel below, there is a significant drop-off:

- Viewed: 1.97M users (100%)
- Added to Cart: 1.29M users (66%)
- Converted: 405K users (21%)

This insight highlights the opportunity to improve conversion by encouraging action after cart addition.

[PLOT 4 - Conversion Funnel Image]



6. System Architecture

Architecture Diagram Placeholder:

- MongoDB → Products, Users, Transactions
- HBase → Sessions
- Spark → Processing & Integration
- Dashboard → Visualizations

7. Limitations and Future Work

The short project timeframe limited deeper experimentation with advanced analytics and additional deployment scenarios.

- Product recommendation logic remains basic; future work could explore collaborative filtering or machine learning models
- The dashboard is accessible only on the local machine and was not hosted online (e.g., Insyt.co) due to time limitations
- The system handled a large dataset with over 10,000 users, 5,000 products, 25 categories, 500,000 transactions, and 2 million sessions. While the performance was stable, future testing could explore how the system behaves when deployed across multiple machines (nodes), such as in a Spark cluster on Hadoop or in a cloud setup like AWS EMR. For example, deploying Apache Spark on a three-node cluster would allow parallel data processing and load balancing, which is useful for very large-scale production environments.

8. Technologies Used

Python: Main language used for all scripts and data handling
- **Pandas:** For data manipulation and analysis
- **Plotly & Dash:** For data visualization and interactive dashboard
- **MongoDB:** For storing document-based collections (users, transactions, products)
- **HBase:** For storing session logs in a wide-column format
- **Apache Spark (PySpark):** For distributed data processing, cohort analysis, and revenue aggregation
- **HappyBase:** For Python integration with HBase

9. References

-**MongoDB Docs:** https://www.mongodb.com/docs
- **Apache HBase:** https://hbase.apache.org/book.html
- **Apache Spark:** https://spark.apache.org/docs
- **Plotly Dash:** https://dash.plotly.com
- **Dataset Generator:** dataset_generator.py
- **GitHub Repository:** https://github.com/niyonsenga1/ecommerce-bigdata-analytics