OLIST E-COMMERCE DATA PIPELINE

Module 2 – End-to-End ELT & Analytics | DuckDB · Parquet · GCS · BigQuery · dbt · Looker Studio / Plotly Dash

AGENDA

- Executive Summary:
  - Purpose, focus and assumption
  - Problem statement & goals
- Data Architecture
- Implementation of ELT pipeline
- Data quality & testing
- Data Insights: Data Analysis with Python and BI Interface: Looker Studio / Plotly Dash
  - Key metrics dashboard
  - E-commerce sales insights
  - Analysis and findings
- Business recommendations
- Phase 2: get ready for operation with Pipeline Orchestration

BUSINESS SETTING

Olist is a Brazilian **e-commerce marketplace**

The challenge:

With thousands of sellers and millions of orders, Olist needs a robust **data platform** to understand customer behavior, seller performance, and logistics bottlenecks etc

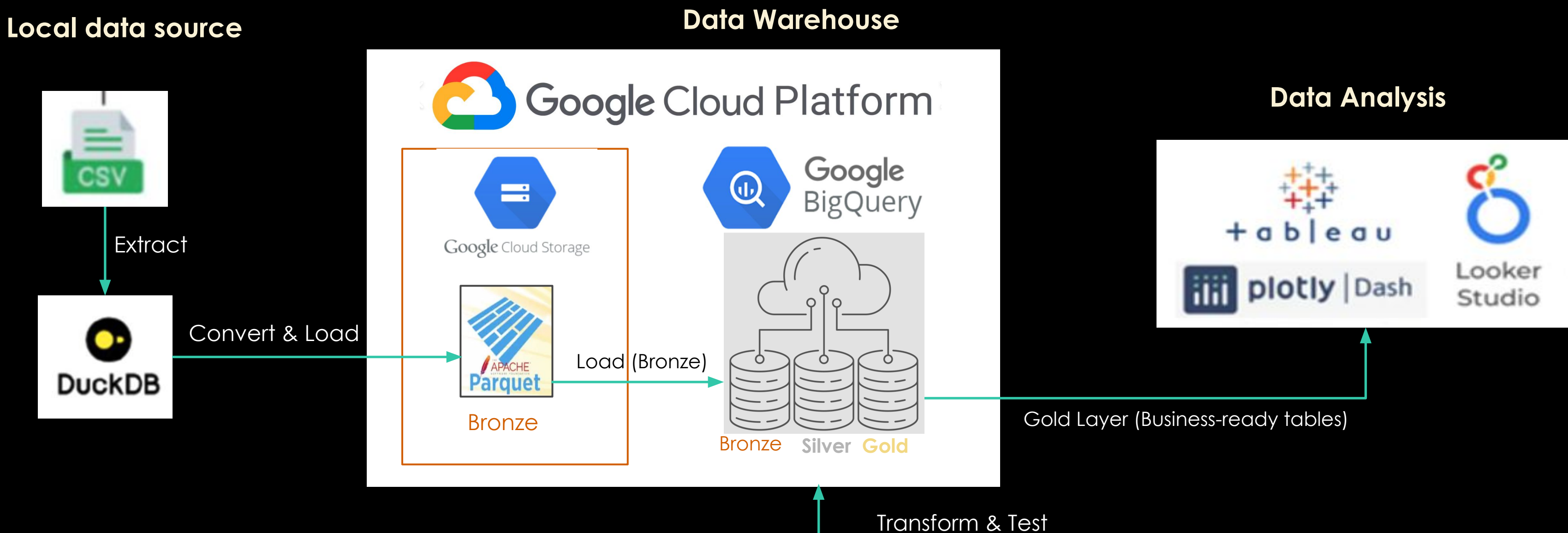
PROBLEM & GOALS

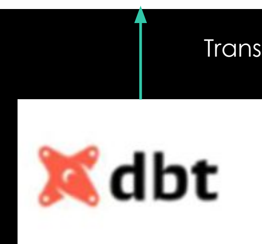
Data given: 9 Olist CSVs raw data available

Outcome required: Transformed into analytics-ready tables & insights in a dashboard

- Reproducible pipeline required: Raw → Marts
- Clean Staging: Renames, Cast Types, Dedupe, Normalization
- EDA and Quality Gates
- Star schema for BI
- BI Dashboards for easy KPI tracking & Business Decisions
- Orchestration to maintain the data in real time (Phase 2)

ARCHITECTURE



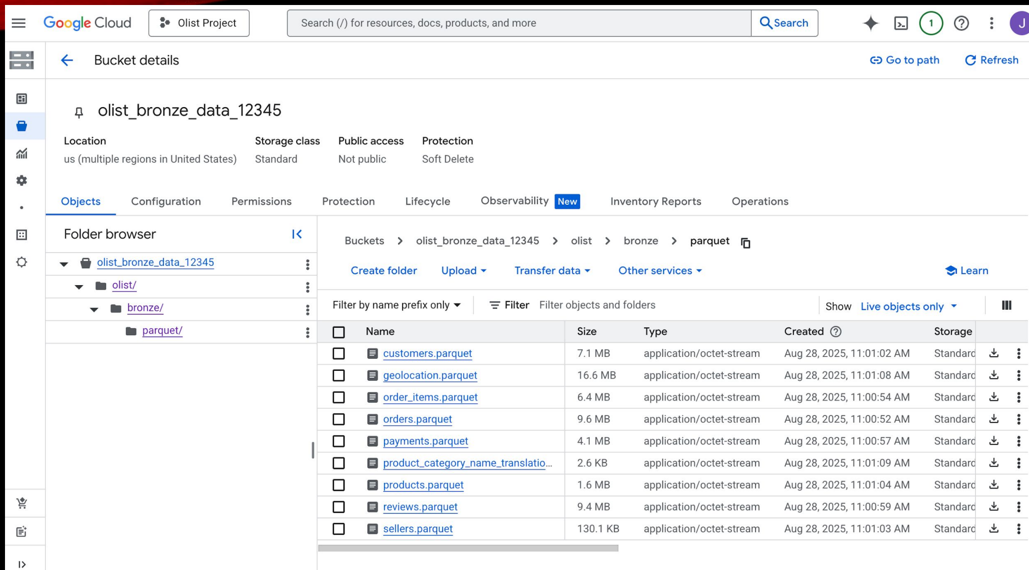


- Start with Kaggle **CSVs** locally. Using **DuckDB**, convert CSVs to **Parquet** (columnar, compressed).
- Those Parquet files are then **uploaded to GCS** as object storage for raw Bronze tables.
- GCS is the middle layer stores Parquet files and act as the source from which BigQuery ingest data.

- dbt takes those raw Bronze tables and **transforms** them into clean, conformed Silver models.
- dbt runs on local machine, Transformations inside BigQuery, which is data warehouse.
- Keep all **transformations in BigQuery** for performance, governance, and BI connectivity.

At this stage, we are working with static Kaggle CSVs

IMPLEMENTATION OF ELT PIPELINE -INGESTION (BRONZE)



- Load CSV into **DuckDB** for type-checking and quick validation
- Convert to **Parquet** for efficient columnar storage and store in **GCS**
- Upload Parquet files to **BigQuery** (`olist_bronze` dataset)

Result: *olist_bronze* dataset in BigQuery with 9 raw files

Why Parquet?

- Compressed, efficient for analytics
- Schema preserved, faster to load into BigQuery
- Scales better than raw CSV

Role of GCS in ingestion

1.Landing zone for Bronze

- Start with Kaggle CSVs locally.
- Using DuckDB/Python, we convert them to **Parquet** (columnar, compressed).
- Those Parquet files are then **uploaded to GCS**.
- At this stage, GCS is acting as your data lake for raw Bronze data.

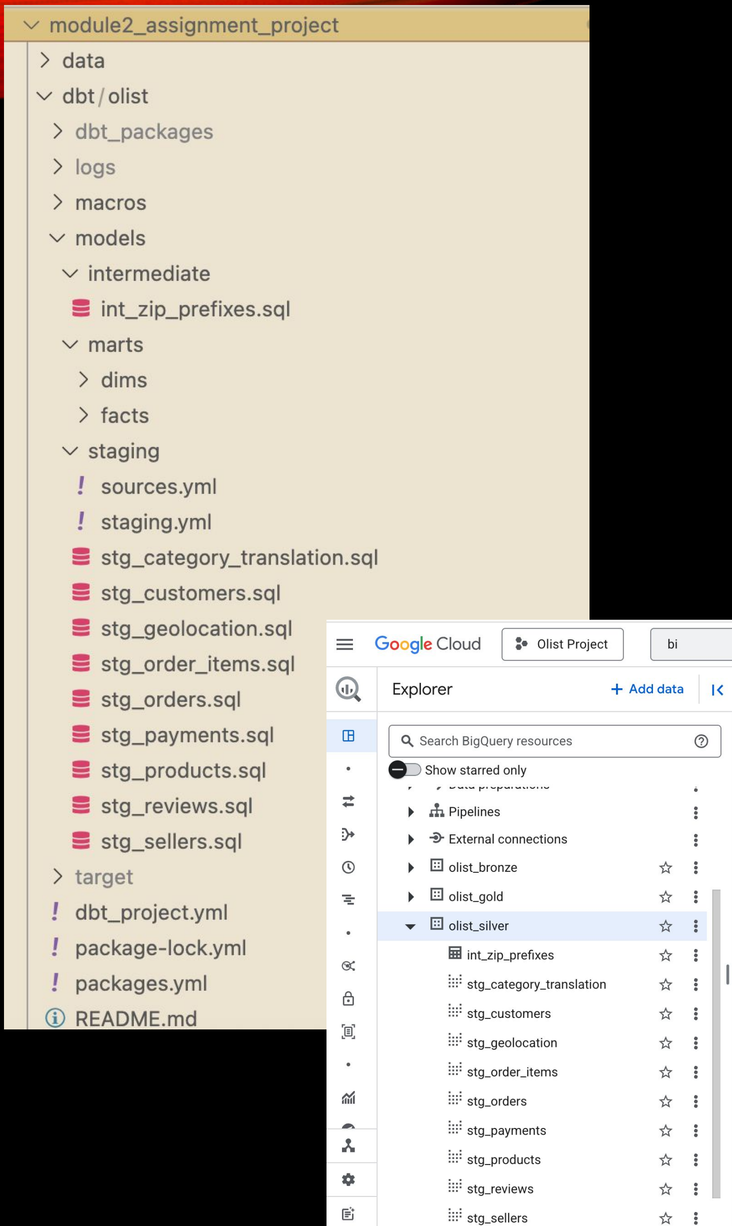
2.Staging point for BigQuery

- BigQuery can't read local **files** directly — it integrates naturally with GCS.
- Our notebook uses the **BigQuery Python client** (or CLI) to LOAD or create Native Tables in BigQuery pointing at those Parquet files in GCS.

3.Benefit: Decoupling compute & storage: GCS stores the data files, BigQuery ingests them

- By keeping Bronze in GCS, we don't overload BigQuery with raw data. If something goes wrong, we can always re-load from GCS.
- We can reprocess/reload from Parquet in GCS at any time without re-downloading the Kaggle dataset.
- It also sets us up for **incremental ingestion** later (partition by date in GCS, load only new files).

IMPLEMENTATION OF ELT PIPELINE –STAGING (SILVER)



9 dbt **staging views** in `olist_silver` and 1 intermediate layer: `int_zip_prefixes` (aggregates geolocation ZIP codes to lat/lng)

Cleaning steps:
- Normalized column names & types
- Deduplication (`stg_reviews`)
- Payment normalization (`stg_payments` → bucket rare values to `other`/`unknown`)
- Provides a clean, consistent interface for downstream models.

Why needed intermediate layer ?

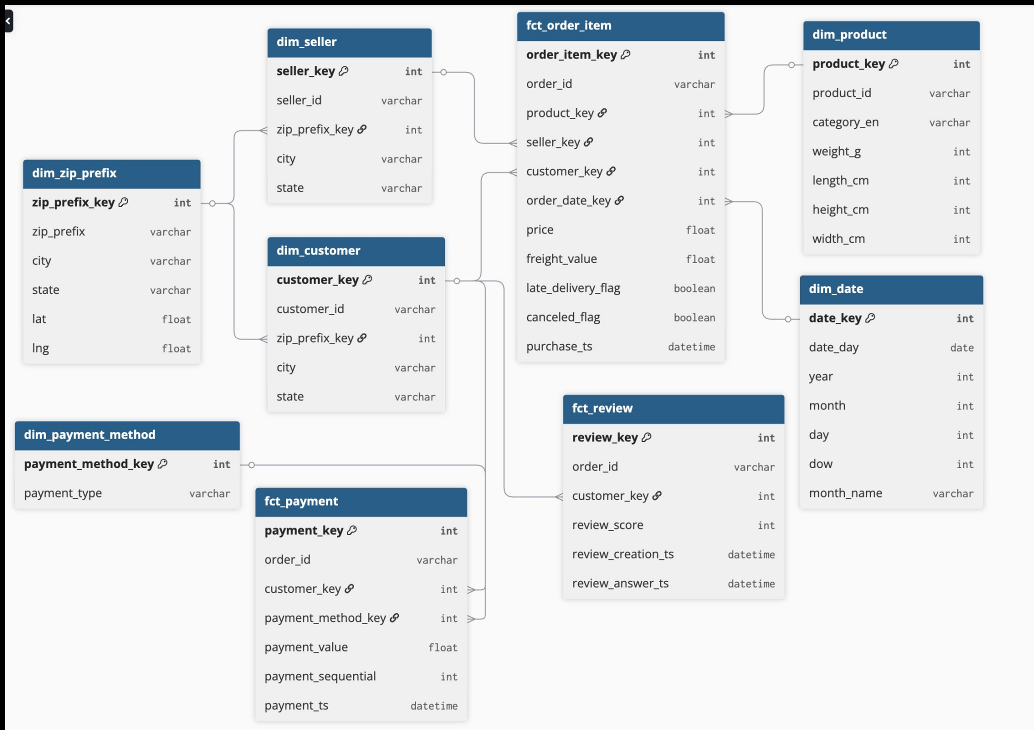
•**Raw geolocation (Bronze)**:
•every customer/seller record comes with a **zip prefix** (lots of repeats, sometimes dirty, inconsistent).

•**Intermediate layer (int_zip_prefixes)**:
•Deduplicates all unique zip_prefix values
•Joins them to their **city, state, lat/lng** from the geolocation dataset
•Produces a **clean reference table** with one row per zip_prefix

- Customers and sellers both point to zip prefixes → instead of repeating city/state in every record, we normalize it.
- This makes **joins simpler and consistent** in Gold (facts/dims).
- Keeps **dim_customer** and **dim_seller** slim and tidy.

👉 So the intermediate layer is just a **clean bridge** between raw zip prefixes and Gold dimensions.

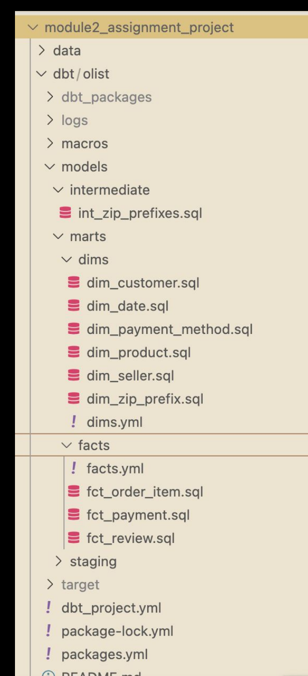
IMPLEMENTATION OF ELT PIPELINE –BUSINESS LAYER (GOLD)

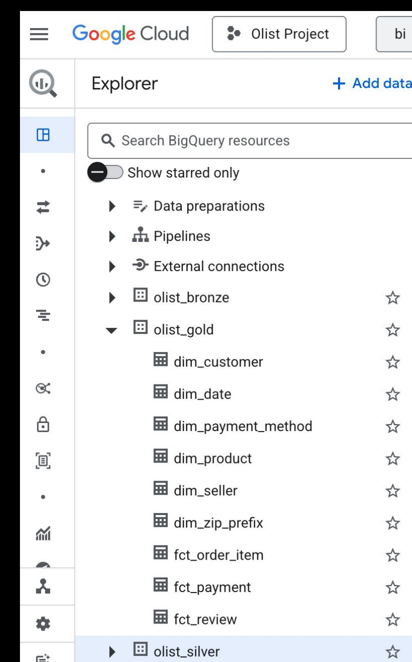


Gold (Business Layer)
6 Dimensions
\`dim_customer\`, \`dim_product\`, \`dim_seller\`, \`dim_zip_prefix\`, \`dim_date\`, \`dim_payment_method\`

3 Facts:
\`fct_order_item\`, \`fct_payment\`, \`fct_review\`

Materialized as **tables** in \`olist_gold\`
Star schema design supports BI dashboards & analytics (refer to star schema in next slide)





IMPLEMENTATION OF ELT PIPELINE – DATA QUALITY & TESTING

Finding (Initial EDA)	Data Cleaning Action
Repeated / inconsistent ZIP prefixes	Built int_zip_prefixes → dim_zip_prefix for normalized reference
Null / invalid payment types	Created dim_payment_method, flagged invalid/null values
Duplicate reviews, skewed ratings	Deduplicated review_id; flagged outliers in review scores
Canceled orders & invalid date sequences	Added canceled_flag, late_delivery_flag; validated date fields
Missing customer/seller location values	Standardized ZIP joins; filled/handled missing city/state
Outliers in freight values	Applied dbt-expectations range tests; flagged anomalies

Ensuring Data Quality & Testing

Data Testing with dbt: **tests** are in YAML files (dims.yml, facts.yml, staging.yml)

Not Null & Unique

Primary keys in dim_customer, dim_product, dim_seller, fct_order_item, etc.

Relationships

Foreign keys like fct_order_item.customer_key → dim_customer.customer_key
fct_order_item.product_key → dim_product.product_key

Accepted Values

payment_type in stg_payments (credit_card, boleto, voucher, debit_card)

dbt-expectations

run expect_column_values_to_be_in_set on payment_type
had tests for late_delivery_flag values (0/1)

✅ Result: A **trusted Silver/Gold layer** with enforced integrity and test coverage for downstream analysis & BI.

BI DASHBOARD 1 (LOOKER STUDIO)

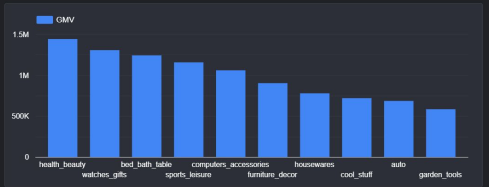
GMV 15,843,553.24	Orders 98,666	Avg Delivery Days 12.41
Late Delivery % 7.74	Items 112,650	AOV (Average Order Value) 160.58

GMV by Month

Orders by Month

Average Delivery by Month

Sales by Product Category (Top 10)



Sales by Customer State

Average Review Score by Month

Review Score Distribution

	review_score	order_id ▾
1.	5	56,699
2.	4	18,944
3.	1	10,721
4.	3	8,029
5.	2	3,049

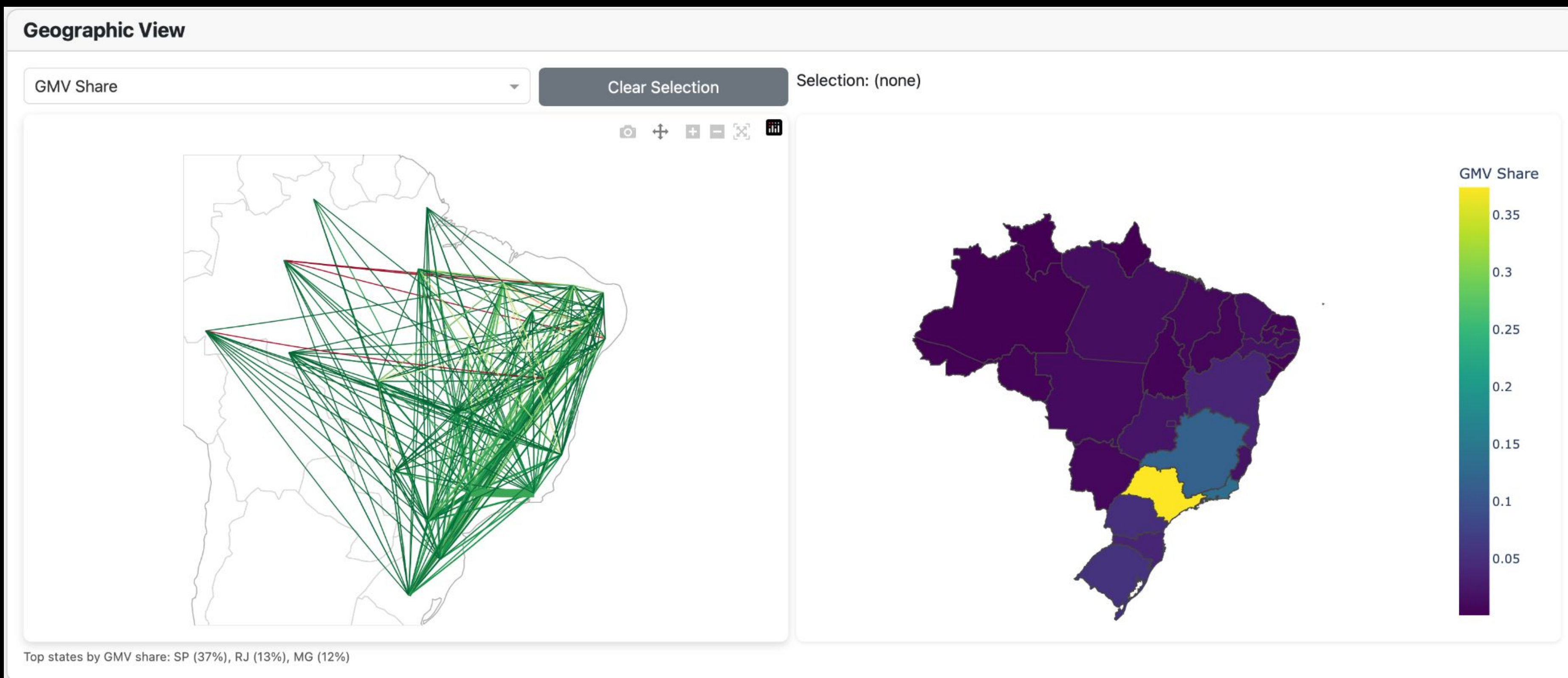
1 - 5 / 6

Top Products Category

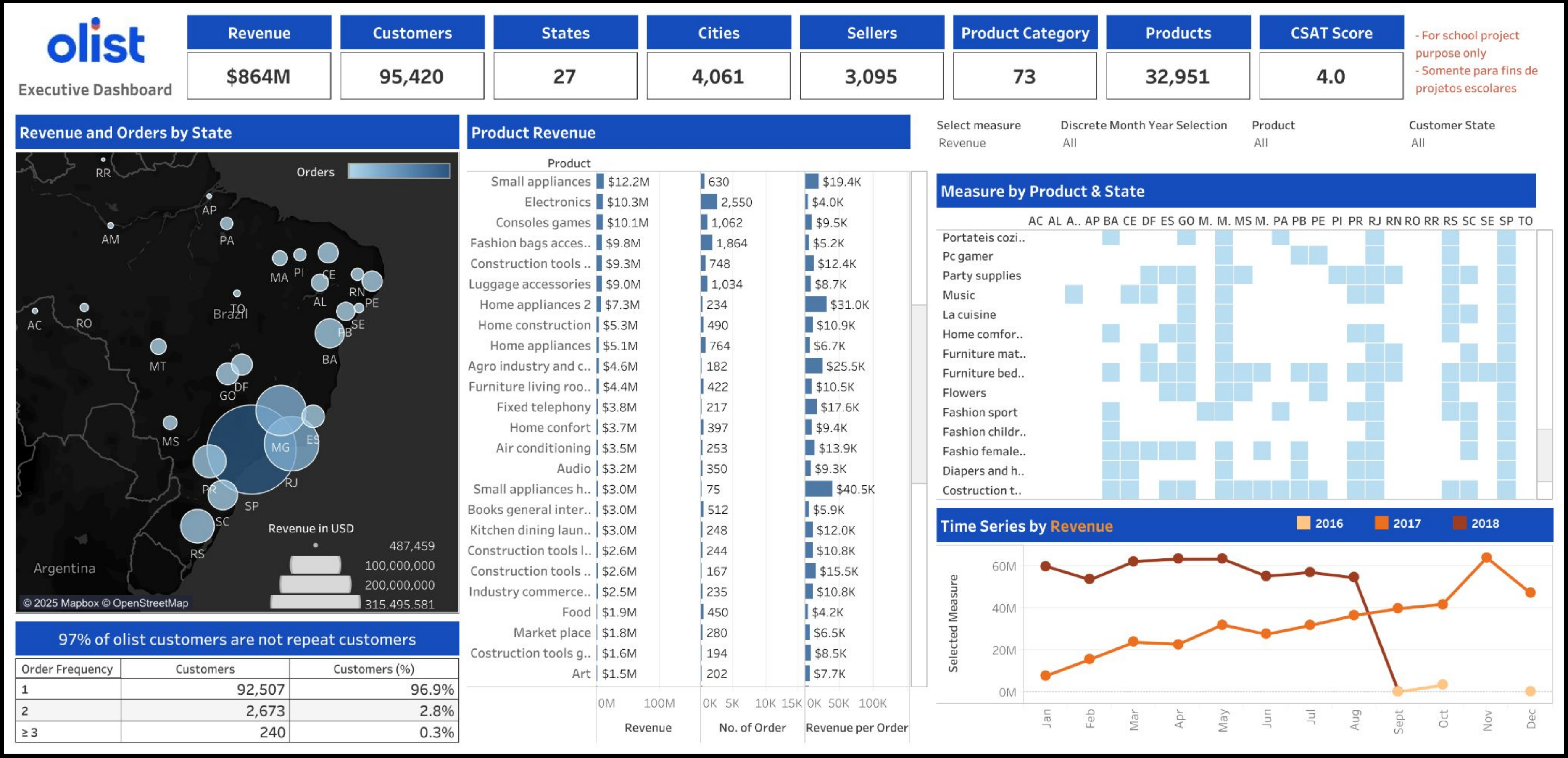
	category_en	Items	GMV ▾	review_score
1.	health_beauty	9,670	1,441,248.07	4.14
2.	watches_gifts	5,991	1,305,541.61	4.02
3.	bed_bath_table	11,115	1,241,681.72	3.9
4.	sports_leisure	8,641	1,156,656.48	4.11
5.	computers_accessories	7,827	1,059,272.4	3.94
6.	furniture_decor	8,334	902,511.79	3.91
7.	housewares	6,964	778,397.77	4.06
8.	cool_stuff	3,796	719,329.95	4.15
9.	auto	4,235	685,384.32	4.07
10.	garden_tools	4,347	584,219.21	4.05

1 - 10 / 74

BI DASHBOARD 2 (PLOTLY DASH)



TABLEAU



Order Frequency	Customers	Customers (%)
1	92,507	96.9%
2	2,673	2.8%
≥ 3	240	0.3%

Product	Revenue	No. of Order	Revenue per Order
Small appliances	$12.2M	630	$19.4K
Electronics	$10.3M	2,550	$4.0K
Consoles games	$10.1M	1,062	$9.5K
Fashion bags acces..	$9.8M	1,864	$5.2K
Construction tools ..	$9.3M	748	$12.4K
Luggage accessories	$9.0M	1,034	$8.7K
Home appliances 2	$7.3M	234	$31.0K
Home construction	$5.3M	490	$10.9K
Home appliances	$5.1M	764	$6.7K
Agro industry and c..	$4.6M	182	$25.5K
Furniture living roo..	$4.4M	422	$10.5K
Fixed telephony	$3.8M	217	$17.6K
Home confort	$3.7M	397	$9.4K
Air conditioning	$3.5M	253	$13.9K
Audio	$3.2M	350	$9.3K
Small appliances h..	$3.0M	75	$40.5K
Books general inter..	$3.0M	512	$5.9K
Kitchen dining laun..	$3.0M	248	$12.0K
Construction tools l..	$2.6M	244	$10.8K
Construction tools ..	$2.6M	167	$15.5K
Industry commerce..	$2.5M	235	$10.8K
Food	$1.9M	450	$4.2K
Market place	$1.8M	280	$6.5K
Costruction tools g..	$1.6M	194	$8.5K
Art	$1.5M	202	$7.7K

INSIGHTS & RECOMMENDATIONS

1) Limited repeat customers
   - Loyalty program to manage customer ecosystem
2) Logistics bottlenecks
   - Partnerships with delivery partners
   - Tracking of service levels
3) 18 Sellers generate 80% of GMV – concentration risk
   - Develop seller ecosystem and incentives to diversify
4) Regional concentration of GMV in 3 states
   - Market study to develop market in rest of states

BUSINESS VALUE

Data-driven approach

- Ability to track KPIs in real time
- Receive feedback from the market in real time
- Provide benchmark for future strategy testing

PHASE 2

When we move to live or updated sources, we will :

- Add Snapshot, allow us to capture changes in slowly changing dimensions (SCD Type 2).
- Use Meltano to manage extract & load into BigQuery.
- Orchestrate Pipeline with Dagster to ensure:
  - Bronze ingestion → Silver cleaning → Gold marts run in sequence
  - Snapshots run before transformations
  - Automated tests + alerts on failures
- Continue connecting gold layer to BI Tools for updated sources