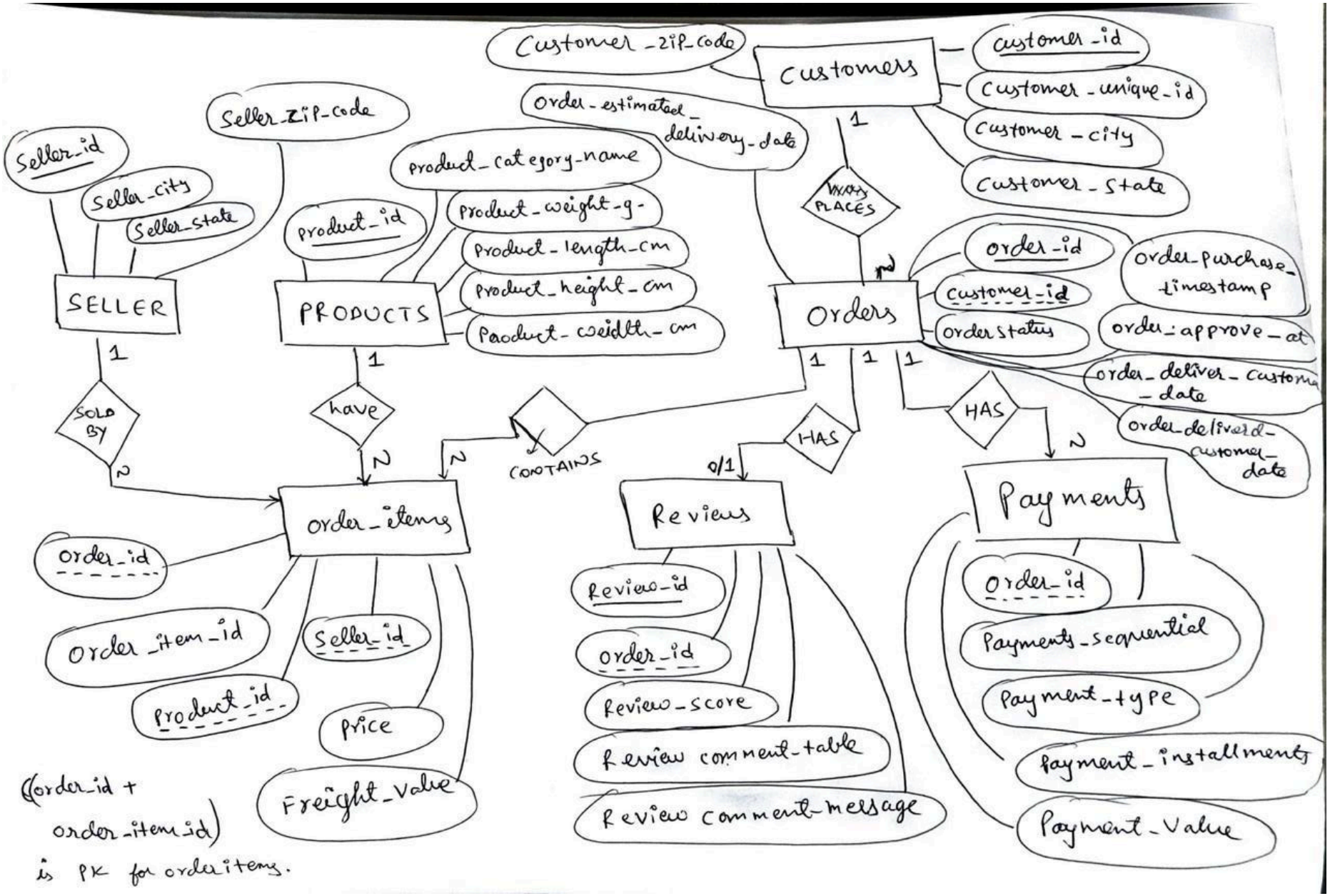
PHASE 1 REPORT

Team members:

1.Jayachandra galda 2.Gayathri Adulla 3.Venkata sai sumanth bijjam

1. Entity Relationship (ER) Diagram

Our database design is based on the Olist Brazilian E-commerce dataset, which contains transactional data from an online marketplace. After analyzing all nine CSV files, we identified seven core entities: Customers, Orders, Order Items, Products, Sellers, Payments, and Reviews.


Customer_ZIP_code
customers
customer_id
Customer_unique_id
Customer_city
Customer_State
Order_estimated_delivery_date
Seller_ZIP_code
Seller_id
Seller_city
Seller_state
product_category_name
product_id
Product_weight_g
Product_length_cm
Product_height_cm
Paoduct_weidth_cm
WHERE PLACES
order_id
order_purchase_timestamp
customer_id
order_approve_at
order status
order_deliver_customer_date
order_delivered_customer_date
SELLER
PRODUCTS
Orders
1
SOLD BY
have
CONTAINS
HAS
HAS
N
0/1
order_items
Reviews
Payments
order_id
Order_item_id
Product_id
Seller_id
price
Freight_Value
Review_id
order_id
Review_score
Review comment_table
Review comment_message
order_id
Payments_sequential
Payment_type
Payment_installments
Payment_Value
(order_id + order_item_id) is PK for order items.

2. Table Descriptions

1. Customers

Stores customer-level information such as location.
Primary Key: customer_id
Attributes include: customer_unique_id, customer_city, customer_state, customer_zip_code_prefix.

2. Orders

Represents each transaction placed by a customer.
Primary Key: order_id
Foreign Key: customer_id → *Customers*
Attributes include:
order_status, order_purchase_timestamp, order_approved_at, order_delivered_carrier_date, order_delivered_customer_date, order_estimated_delivery_date.

3. Order Items

Breaks each order into individual product units.
Composite Key: (order_id, order_item_id)
Foreign Keys: order_id → Orders, product_id → Products, seller_id → Sellers
Attributes include: price, freight_value.

4. Products

Catalog of products sold on the platform.
Primary Key: product_id
Attributes include:

product_category_name, product_weight_g,
product_length_cm, product_height_cm, product_width_cm.

5. Sellers

Stores seller business and location details.
Primary Key: seller_id
Attributes include:
seller_city, seller_state.

6. Payments

Represents customer payments for each order.
Composite Key: (order_id, payment_sequential)
Foreign Key: order_id → Orders
Attributes include:
payment_type, payment_installments, payment_value.

7. Reviews

Customer feedback on orders.
Primary Key: review_id
Foreign Key: order_id → Orders
Attributes include:
review_score, review_comment_title, review_comment_message.

Columns intentionally excluded from schema (not useful for analysis):

- review_creation_date (Reviews)
- product_name_length (Products)

- product_photo_qty (Products)
- product_description_length (Products)
- shipping_limit_date (Order Items)

3. Normalization & 3NF Justification

1NF Compliance

- Every table contains atomic values.
- No repeating groups.
- Each row represents a unique instance (customer, order, order item, etc.).

2NF Compliance

- No partial dependencies exist.
- In order_items, the composite key is (order_id, order_item_id), and all non-key attributes depend on the full composite key.
- In payments, the composite key is (order_id, payment_sequential), and non-key attributes depend on both fields.

Since no non-key attribute depends on *part* of a composite key → 2NF satisfied.

3NF Compliance

- No transitive dependencies exist in any table.

Examples:

- Customer city/state depend ONLY on customer_id — not on orders.
- Product dimensions and category depend ONLY on product_id — not on order_items.
- Review text and score depend ONLY on review_id, not on order details.
- Payments attributes (type, installments, value) depend ONLY on the composite key and not on indirect attributes.

Therefore:

All non-key attributes depend only on the primary key.

No attribute depends on other non-key attributes
No redundancy or update anomalies exist

Schema is fully in 3NF.