DATA PREPARATION & EDA

1. Business Processes

The business process underlying this project is the **end-to-end delivery workflow** of Ahamove's on-demand logistics platform, which connects customers who need delivery services with a network of active drivers (suppliers) operating across multiple cities in Vietnam. The system functions as a real-time digital marketplace, where supply (drivers) and demand (customer orders) are continuously matched through an intelligent dispatch algorithm.

When a **customer creates a delivery request** on the Ahamove application, the system records the order information, such as pickup and drop-off locations, service type, and delivery distance, and immediately initiates a **matching process** to assign the request to an available driver nearby. This initiates the **order lifecycle**, which can be divided into several key operational stages:

Order Creation (Request Stage): A customer submits a new order request through the Ahamove app or API. The system generates a unique order identifier (order_id) and logs the event as create_time.

Driver Matching and Acceptance: The platform's dispatching engine sends the request to one or more nearby drivers based on factors such as distance, availability, and performance score. When a driver accepts the order, the system updates the record with an accept_time timestamp and assigns the corresponding supplier_id.

Pickup and Delivery Process: After acceptance, the driver proceeds to the pickup location (board_time and pickup_time) to collect the package. Once the delivery is completed, the system records the complete_time and updates the order status to COMPLETED.

Cancellation or System Timeout: If no driver accepts the request within a defined time threshold, the system may automatically cancel the order (cancel_time, cancel_by_user, or cancel_code), or the customer may manually cancel it from the app. These events are critical indicators for analyzing the **Cancellation Rate (CR)** and the **Acceptance Rate (AR)**.

Performance Tracking and KPI Logging: Each event within the order lifecycle is stored in Ahamove's centralized database, which aggregates performance

metrics such as **Fulfillment Rate (FR)**, **Cancellation Rate (CR)**, **Productivity**, and **Lead Time**. These KPIs provide essential insights into the efficiency of the delivery network and driver engagement.

This entire business process operates **continuously and in real time**, with thousands of transactions per day across different cities. Each action—whether initiated by a **user**, a **driver**, or the **system itself**, is captured automatically through event logs and translated into structured data records within Ahamove's data warehouse.

Understanding this business process is essential for interpreting the data model and analytical results in later sections. It clarifies the logical relationships between customer behavior, driver activity, and system operations, forming the foundation for evaluating performance efficiency and identifying areas for operational improvement.

2. Data Source

The dataset used in this project originates from **Ahamove's internal operational database**, which continuously records activities from three main entities: **drivers (suppliers), users (customers), and the system itself**. Every data point in the **Orders** table is generated in **real time**, reflecting dynamic interactions between these entities throughout the lifecycle of a delivery order.

When a customer creates a new request on the Ahamove platform, the system immediately generates a unique **order_id** and begins tracking multiple events, such as request creation, driver acceptance, pickup, delivery, or cancellation. Each of these actions is logged as an activity within Ahamove's centralized database. Similarly, drivers' operational behaviors (e.g., accepting, rejecting, or completing orders) and automated system responses (e.g., timeout cancellations or reassignment) are also captured as structured event data streams.

After being generated in real time, these raw event records are **automatically processed and standardized** by Ahamove's internal systems before being stored in a unified **relational database**. This ensures that timestamps, identifiers, and categorical variables follow a consistent schema across different data sources. As a result, each field in the **Orders** table (such as create_time, accept_time, pickup_time, and complete_time) represents the precise moment a specific activity occurred in the

delivery workflow, while other fields (e.g., status, cancel_by_user, cancel_code, distance) describe the operational context and outcome of each transaction.

The **Suppliers** table is derived from the same system but focuses on driver-level attributes, including account creation, activation, last activity, and demographic information such as age. These details are also updated continuously based on the driver's engagement on the platform.

All data were **collected and extracted directly from Ahamove's production environment** via the company's internal data integration system, ensuring high reliability and data freshness. Once retrieved, the datasets were imported into **Microsoft Fabric's Lakehouse environment**, where they served as the foundation for subsequent transformation, modeling, and visualization processes. This real-time, system-generated data collection mechanism guarantees that the analysis reflects **actual operational behaviors** rather than sampled or simulated data, thereby enhancing the credibility and applicability of the project's findings.

3. Data Description

In this project, the data utilized consist of two main tables: **Orders** and **Suppliers**, both sourced from Ahamove's internal system.

The **Orders** table provides detailed information about individual orders, including their status, processing times, and delivery-related details. The **Suppliers** table contains information about delivery personnel, including account activation times, activity logs, and age.

The following tables summarize the columns in each dataset:

Table 1: Orders (2,256,118 rows × 21 columns)

Column name	Data type	Data description
Unnamed: 0	int64	Index column (not used for analysis)
create_time	object	Timestamp when the customer created the request
order_time	object	Timestamp when the order was recorded in the system

order_date	object	Date of the order placement
accept_time	object	Timestamp when the driver accepted the order in the app
board_time	object	Timestamp when the driver started processing the order
pickup_time	object	Timestamp when the driver began the pickup/delivery
complete_time	object	Timestamp when the order was completed
cancel_time	object	Timestamp when the order was cancelled
order_id	object	Unique order identifier
stop_id	object	Identifier of the delivery stop
service_id	object	Service type code
supplier_id	float64	Driver identifier
actual_city_name	object	City where the order was executed
district	object	District or administrative area
status	object	Order status (e.g., COMPLETED, CANCELLED)
distance	float64	Delivery distance (in kilometers)
stop_status	object	Status of the delivery stop
cancel_comment	object	Reason for cancellation provided by the user
cancel_by_user	object	Indicator whether the order was cancelled by the user
cancel_code	object	Cancellation reason code

Table 2: Suppliers (15,196 rows × 9 columns)

Column name	Data type	Data description
Unnamed: 0	int64	Index column (not used for analysis)
id	float64	Driver identifier
activate_time	object	Timestamp when the driver activated their account
create_time	object	Timestamp when the driver account was created
last_activity	object	Timestamp of the driver's last interaction with the app
last_login	object	Timestamp of the most recent login
first_complete_time	object	Timestamp when the driver completed their first order
first_activate_time	object	Timestamp of the driver's first account activation
age	float64	Driver's age

4. EDA

Exploratory Data Analysis was conducted using Python, primarily leveraging the following libraries: (1) **pandas**: for data manipulation and inspection; (2) **numpy**: for numerical operations; (3) **matplotlib.pyplot** and **seaborn**: for data visualization. The purpose was to understand the distribution, missing values, duplicates, and overall quality of the Orders dataset before performing further analyses.

4.1. Orders

4.1.1. Missing values

Missing values were examined using order_df.isnull().sum() in Python. The columns with missing values are:

Table 3: Missing Values Analysis for the Orders Table

Column	Missing Count	Missing Rate (%)
cancel_code	1,639,010	72.65%
cancel_comment	1,521,206	67.43%
cancel_time	1,511,358	66.99%
stop_status	1,500,373	66.50%
cancel_by_user	1,429,084	63.34%
complete_time	744,628	33.00%
pickup_time	744,354	33.00%
board_time	649,314	28.78%
supplier_id	613,02	27.17%
accept_time	563,226	24.96%

Analysis of Missing Values

- **Cancellation Fields (High Missing Rate):** Columns related to order cancellation (cancel_time, cancel_comment, cancel_by_user, cancel_code) exhibit the highest missing rates, ranging from 63% to 72%. This pattern is operationally expected, as these fields are only populated for orders that were cancelled. For successfully completed orders (status = COMPLETED), these columns naturally contain null values.
- **Processing Time Fields:** Columns tracking the physical delivery process (accept_time, board_time, pickup_time, complete_time) have missing rates between **24%** and **33%**. These values are absent only when an order was cancelled before reaching the corresponding stage (e.g., cancelled before driver acceptance).

- **Driver ID (supplier_id):** The supplier_id is missing for approximately **27%** of orders, which is consistent with business operations, as these represent orders that were either not assigned to a driver or were cancelled before acceptance.

Overall, the observed missing values align with the end-to-end delivery workflow and do not indicate data quality issues. Rather, they provide meaningful insight into the lifecycle status of the 2,256,118 orders.

4.1.2. Duplicated Data

Duplicate rows were checked using order_df.duplicated().sum(). The Orders table contains **no duplicate records**, ensuring that each row represents a unique order. Any duplicates would be removed to prevent bias in subsequent analyses.

4.1.3. Quantitative Variable

In the Orders table, the only quantitative variable of interest is **distance** (delivery distance).

Table 4: Statistic of distance

Statistic	distance (km)
Count	2,256,118
Mean	7.03 km
Std	4.41 km
Min	0.00 km
50% (Median)	7.12 km
75% (Third Quartile)	10.00 km
Max	39.87 km

The average distance is **7.03 km**, ranging from **0 to 40 km**, indicating a **right-skewed distribution**. This reflects the operational reality that most orders are short-distance deliveries, while a smaller proportion involve longer distances.

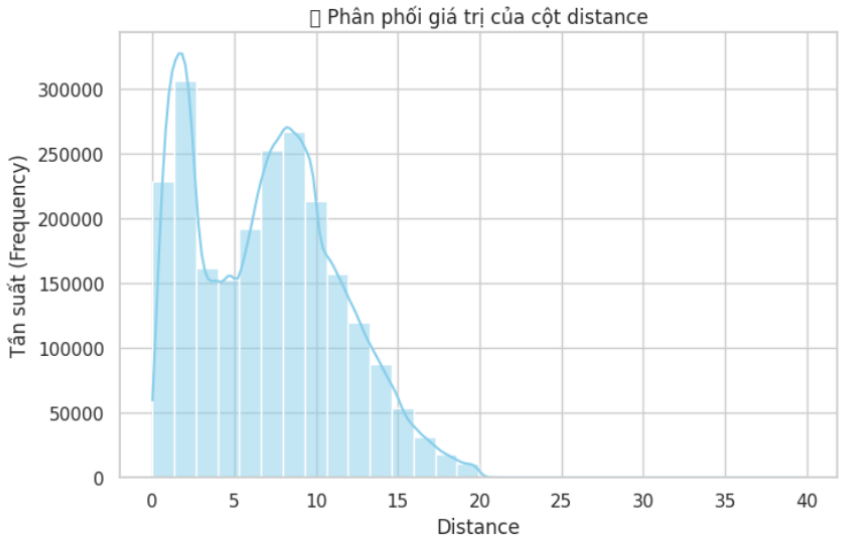

Figure 1: Bar chart showing the distribution of the distance variable.

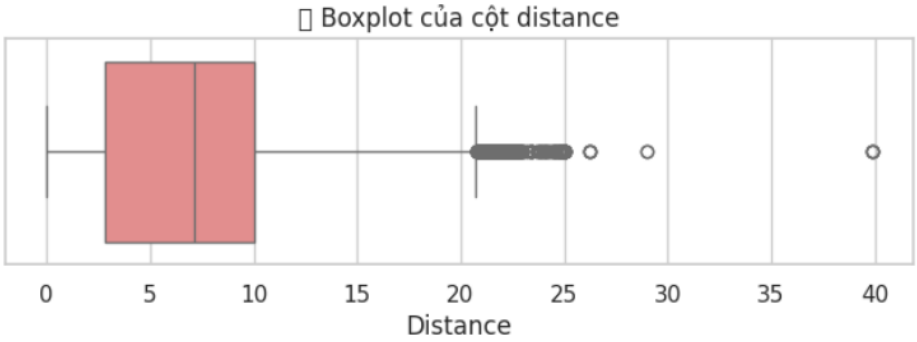

Figure 2: Boxplot illustrating the distribution of the distance variable.

The frequency distribution (top) and the Boxplot (bottom) highlight the distribution characteristics of distance:

- **Multimodal Distribution:** The frequency chart shows a **multimodal** (not normal) distribution with two distinct peaks: one at a very short distance (approximately 2-3 km) and another around a mid-range distance (approximately 8-10 km). This pattern strongly reflects Ahamove's **diverse service structure**, which encompasses both short-haul/hyperlocal deliveries and longer-distance traditional packages.
- **Right-Skewness:** The distribution is right-skewed, indicating that the majority of orders fall within the short-to-medium range, with a smaller proportion of orders requiring longer distances (beyond 20km).
- **Boxplot Confirmation:** The Boxplot confirms that 75% of all orders are delivered within **10km** (the third quartile). While outliers extend up to almost

40km, this range is **operationally realistic** for last-mile logistics and is not treated as data error but as necessary business data for long-distance performance analysis.

In conclusion, the Orders data exhibits high quality (no duplicates, no missing core identifiers), and the distance distribution accurately reflects the platform's varied operational characteristics.

4.2. Suppliers

The Suppliers dataset comprises **15,196** rows and **9** columns, providing detailed information about the drivers (suppliers) active on the platform.

4.2.1. Missing values

Missing values were examined using supplier_df.isnull().sum(). The columns with missing values are summarized in **Table 3.3.2a**:

Table 5: Missing Values in Suppliers Dataset

Column	Missing Count	Missing Rate (%)
activate_time	10,124	66.6%
id	1	< 0.01%
create_time	1	< 0.01%
last_activity	1	< 0.01%
last_login	1	< 0.01%
first_activate_time	1	< 0.01%
age	4	< 0.01%
first_complete_time	4	< 0.01%

activate_time (10,124 missing values): This is the most significant observation, indicating that over 66% of accounts created (create_time is mostly complete) never completed the activation process required to officially begin accepting orders. This suggests a potential bottleneck in the conversion process from account registration to active driver status, which is a critical factor for analyzing supply capacity. Core

Identification Columns (id, create_time, etc.): These columns are almost entirely complete, with only 1 missing value. This single record (where id is missing) appears to be entirely null across several time columns and should be removed to ensure data integrity.

4.2.2. Duplicated Data

The check for duplicate rows using supplier_df.duplicated().sum() confirmed that the **Suppliers** table contains **0** duplicate records. This confirms that each row represents a unique driver, which is essential for accurate calculations related to individual productivity and resource allocation without introducing bias.

4.2.3. Variable Analysis

Quantitative Variables

The primary quantitative variable of interest is **age**. Descriptive statistics for numerical columns are shown in **this table**:

Table 6: Descriptive Statistics for Numerical Columns in Suppliers

Statistic	age (Years)
Count	15,192
Mean	30.67
Std	11.63
Min	6.00
Max	1016.00
50% (Median)	29.00

Descriptive statistics show the mean driver age is **30.67** years, with a standard deviation of 11.63 and a median of 29.00. This distribution indicates that the active driver pool is relatively young and concentrated in the 20-40 age bracket. However, the age column exhibits **clear outliers**, with a minimum value of 6.00 and a maximum

value of 1016.00. The age distribution chart visually confirms the high concentration at lower age values and the presence of these extremely large, anomalous values extending far beyond any logical human age. These extreme values are indicative of **data entry errors** and must be addressed (e.g., capped or removed) before any reliable statistical analysis concerning driver demographics can be performed.

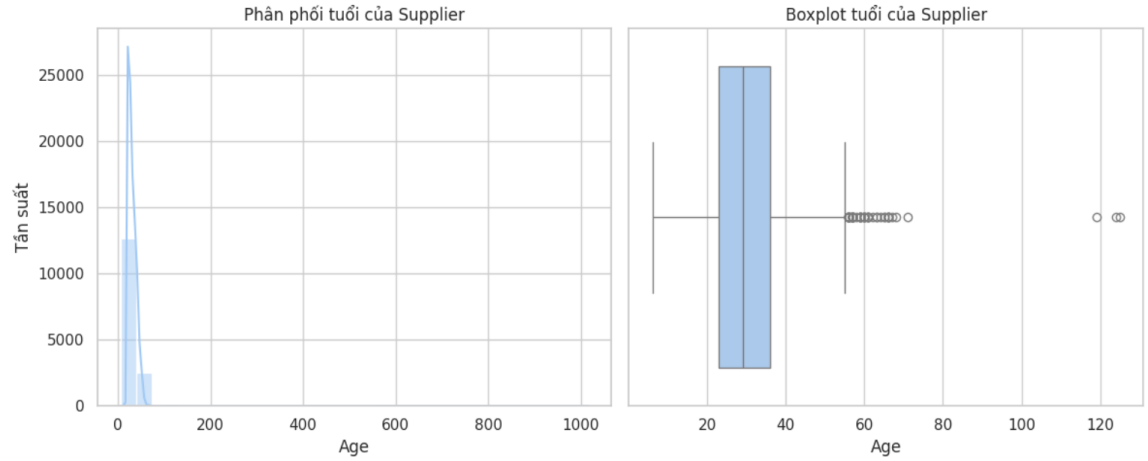

Figure 3: Distribution and Boxplot of Driver Age

Figure 3 clearly illustrates the highly right-skewed nature of the age distribution. The frequency distribution chart (left) confirms the high concentration of data around ages under 100, while the x-axis is elongated due to the presence of extremely large outliers. The Boxplot (right) further emphasizes this: the majority of the data is tightly concentrated around the median (29 years), while a significant number of outliers are visible beyond the upper boundary, confirming the existence of data entry errors or anomalous data points (e.g., ages up to 120 and 1016). These outlying values must be addressed (e.g., capped or removed) prior to conducting reliable statistical analysis.

In summary, while the Suppliers data is clean regarding duplicate records and the completeness of primary identifiers, it requires specific pre-processing steps to address the high missing rate in the activate_time column and the significant outliers present in the age column.