Metrocar Funnel Analysis

Masterschool Mastery Project
Ganna Tsatskina
8 July 2023

Summary

The goal of this project is analysing the customer funnel of Metrocar, a ridesharing app, to identify friction points through the customer journey where users are *struggling* to get the next step.
The funnel model consists of following stages:
Download app → signup → ride request → ride accept → ride complete → payment → review.
Our funnel looks quite healthy. Based on the analysis, the following conclusions were drawn:

- Focus on improving transition from app download to signup and from ride accept to ride complete (lowest conversion rate).

- Concentrate marketing efforts on the IOS platform, which has the largest user base and potential through the funnel.

- Target customers are mainly in the age cohort of 25-34 and 35-44 years old. Important note: our app allows customers not to sign age. So, we got 35% users without age identification.

- Quantity of customers and revenue increased evenly from January 2021 till December 2021, where reached a peak, and then began to decline until the end of observed period (April 2022). It is necessary to find out reasons of the fall.
  Metrocar app usage was not affected by the day of the week.

- Implement surge pricing during peak demand hours between 15:00 and 21:00

- Improve conversion rates at the ride accept and ride complete stages to increase potential revenue by 35% and 41% respectively.

Data and Tools overview

We explored, cleaned and prepared dataset which contains data about users and rides during the period from January 2021 to April 2022. Our database contains further tables:

- **app_downloads**: contains information about app downloads
- **signups**: contains information about new user signups
- **ride_requests**: contains information about rides
- **transactions**: contains information about financial transactions based on completed rides:
- **reviews**: contains information about driver reviews once rides are completed

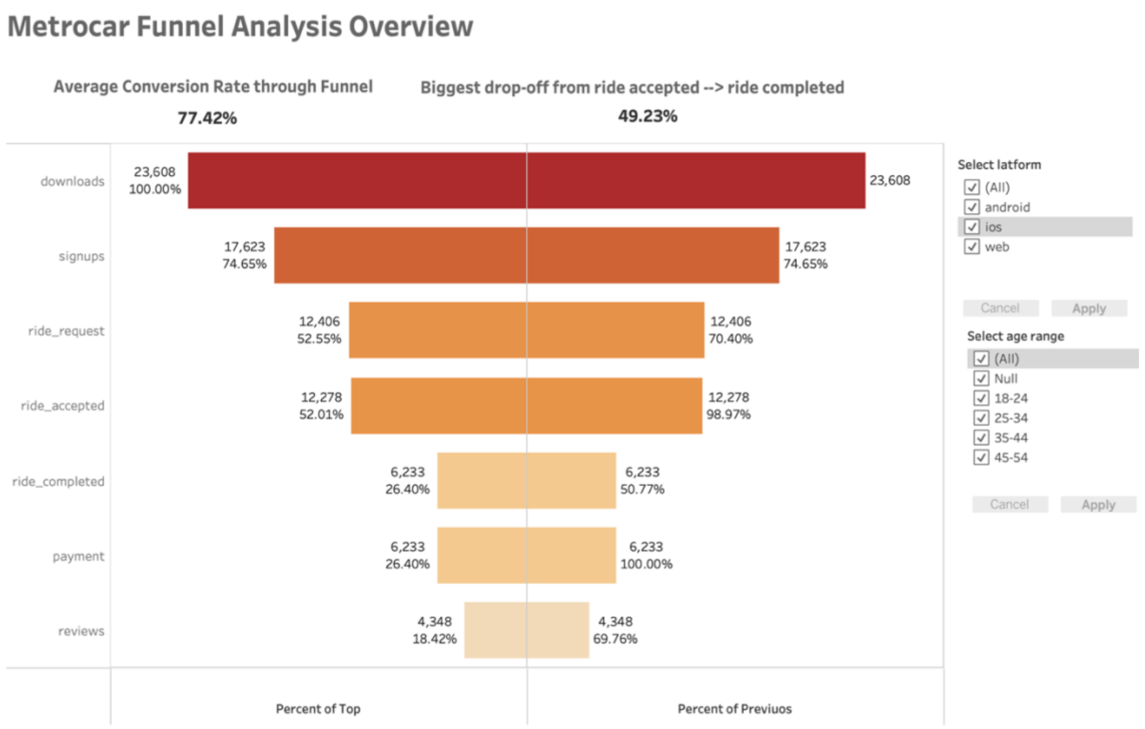
  The data was further analyzed using SQL, simplifying queries with common table expressions, we aggregated the data as follows:

- calculated totals of distinct users_id for following stages - download, signup, ride requested, ride accepted, ride completed, approved transactions, reviews. For this calculation we used COUNT() function and CASE WHEN .. expression;
- distributed the data according to funnel stages, using cohorts of customers platform and age range groups and prepared final table with necessary fields (funnel_name, platform, age_range, download_dt, user_count);
- Saved the final table as .csv file
(Appendix 1)

Further analysis was provided with Google Sheets.
Visualizations were made with Tableau tool.

Overall Funnel Analysis

Aggregated data allows us to analyze it with few levels of granularity: users level, rides level, platforms, age range and download date**.**



Picture 1 Metrocar Funnel

- 75% of the 23608 customers who downloaded app signed up in it. Here we lost 25% of users

- 70% of them requested ride meaning 47.5% of downloads did not make ride request

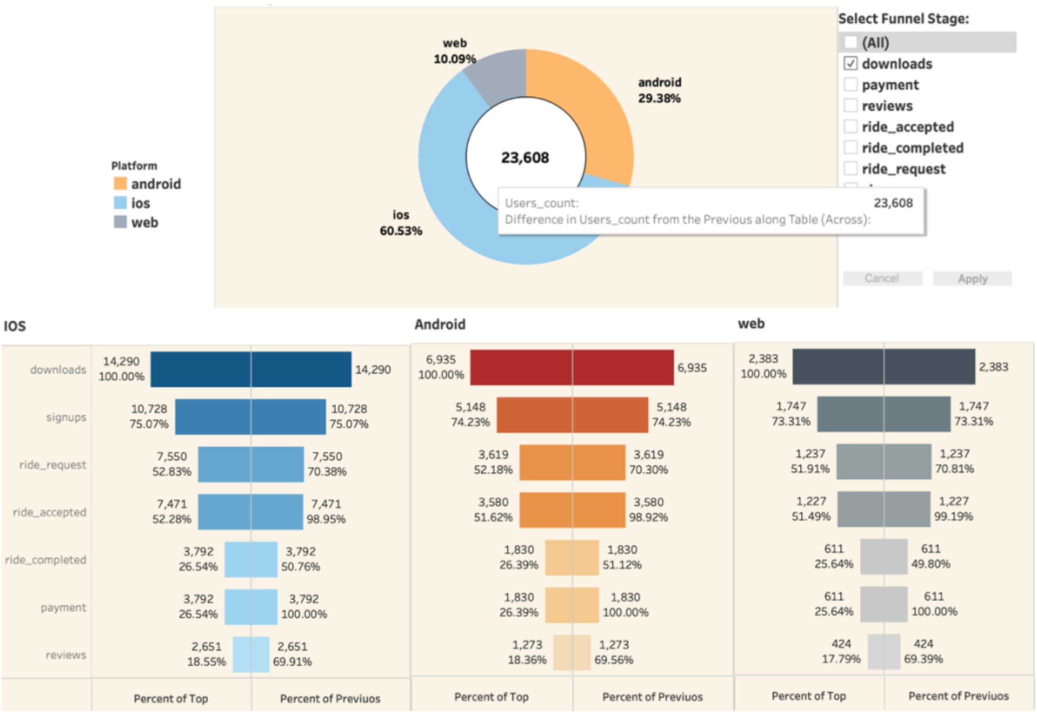
- 99% of requested rides were accepted meaning perfect conversion rate for our funnel

- 51% of requested rides were completed. Here we met specific drop-off points preventing 74% of downloaded app users from next step of the funnel (payment)
- All completed rides where paid
- 70% of customers who completed rides made review meaning 18% from the first stage.

The worst conversion we observe on the last stage (Pic 1). But for purposes of getting income, I propose focus on the other point. Running this on our data, we see that the majority of customers are leaving the funnel after ride was accepted, dropping off before they actually realize riding. 49.23 is the percentage of users who request a ride but cancel before the driver arrives.
This high drop-off rate is an indication that somewhere between these steps, customers are encountering points of friction that are worth investigating.

Platform granularity

Metrocar currently supports 3 different platforms: ios, android, and web. We made further analysis creating donut chart to define distribution of our customers with different platforms (Pic. 2) and three funnels for every platform.



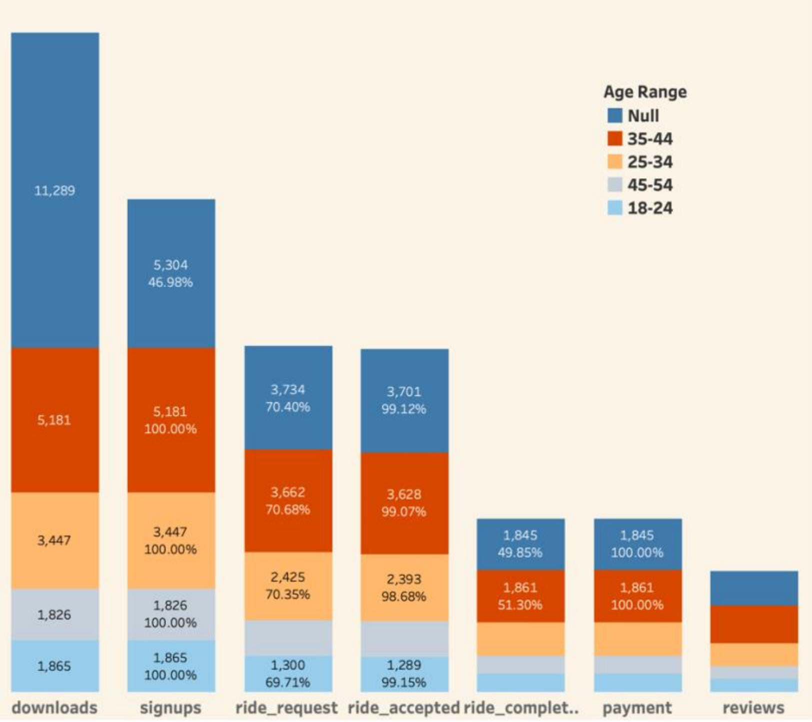
Picture 2. Platform Granularity

All platforms have similar own funnels.
According to the analysis I could recommend focus our marketing budget on the customers who used ios platform as it is largest (60.5% downloads) among our users.

Age Range Granularity

Aggregated table allows us to distribute every stage of funnel on the age ranges. I used bar chart to perform with different colors each age group on the stages of the funnel.

Our users have option do not sign their age. So, 47% users who downloaded application did not mention their age (Pic 4.). This point makes struggle in proper conclusions about this level of granularity.



Picture 3. Age Range Groups Granularity

- What age groups perform best at each stage of our funnel? Which age group(s) likely contain our target customers?

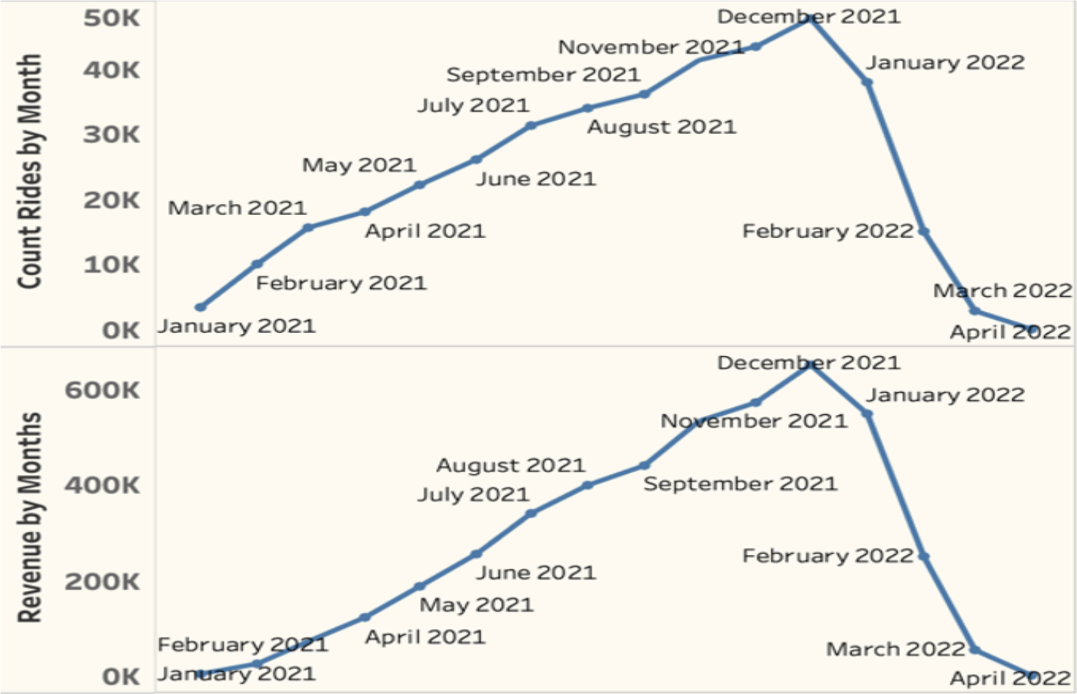
Observing other age groups easy to conclude that our target customers are in the two age groups: largest group – 35-44 years old, second one – 25-34 years old.

Seasonality Observation. Surge Pricing

Primary Business Goal is to increase revenue. One way to realize this goal is urge pricing when there is the greatest demand.

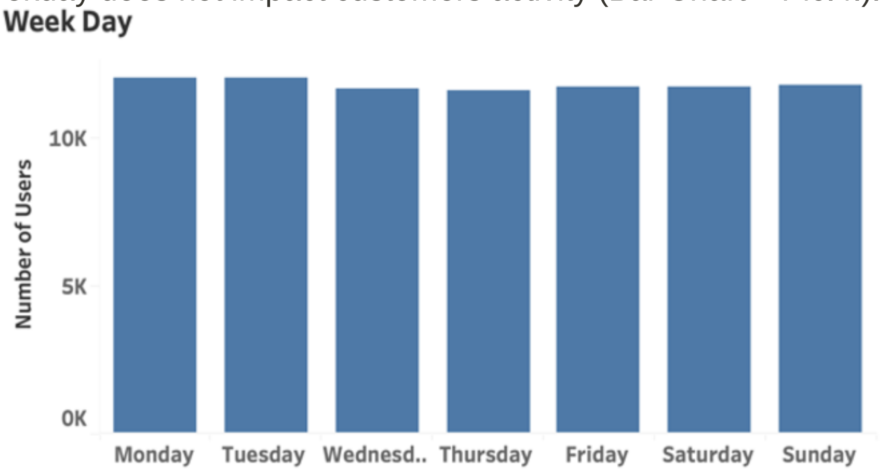
To define points for increasing price we made analysis of our data in different time intervals of time: months, weeks, daytime.

Line chart shows us that our business (number of customers and revenue) increased evenly from January 2021 till December 2021, reached pick, and began fall until the end of observed period (April 2022).



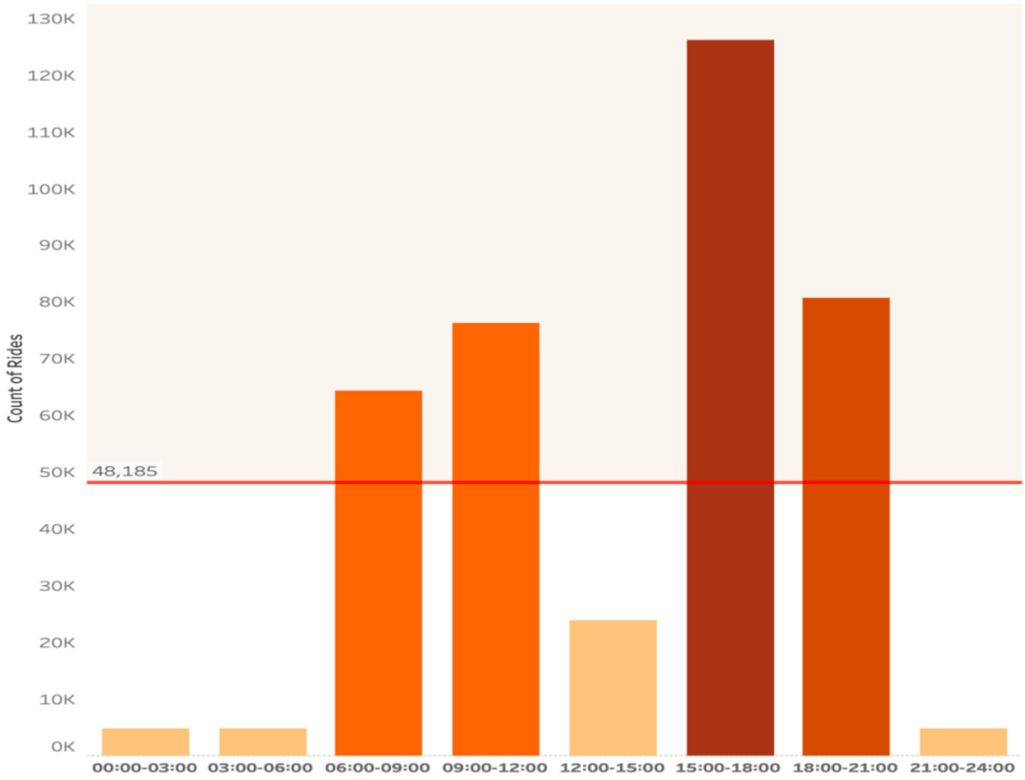
Picture 4. Monthly changes

Weekday does not impact customers activity (Bar Chart – Pic.4.).



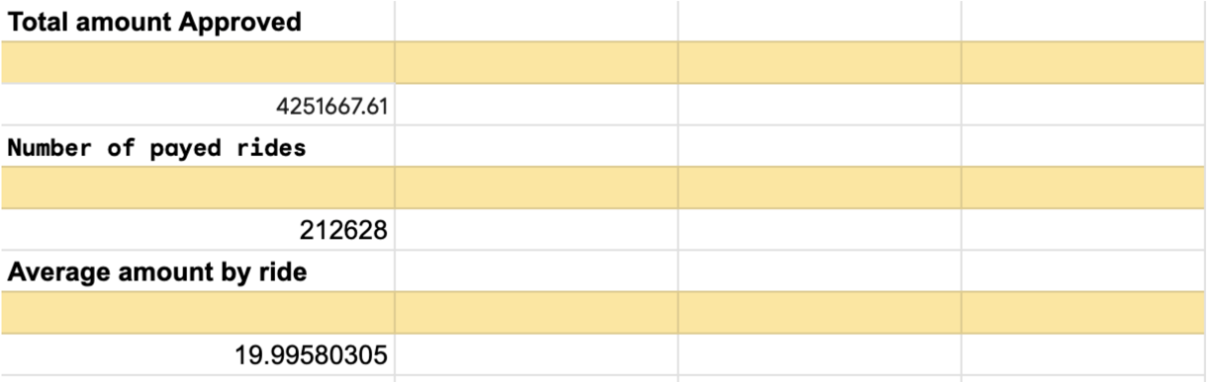
Picture 5. Weekdays rides

The point where we can change prices is daytime demand. To define what part during the day could be in our focus I divided it on the bins three hours each (Pic. 6). Bar Chart with average line shows that the largest demand was between 15:00 and 18:00 also between 18:00 and 21:00 .



Picture 6. Daytime Demand

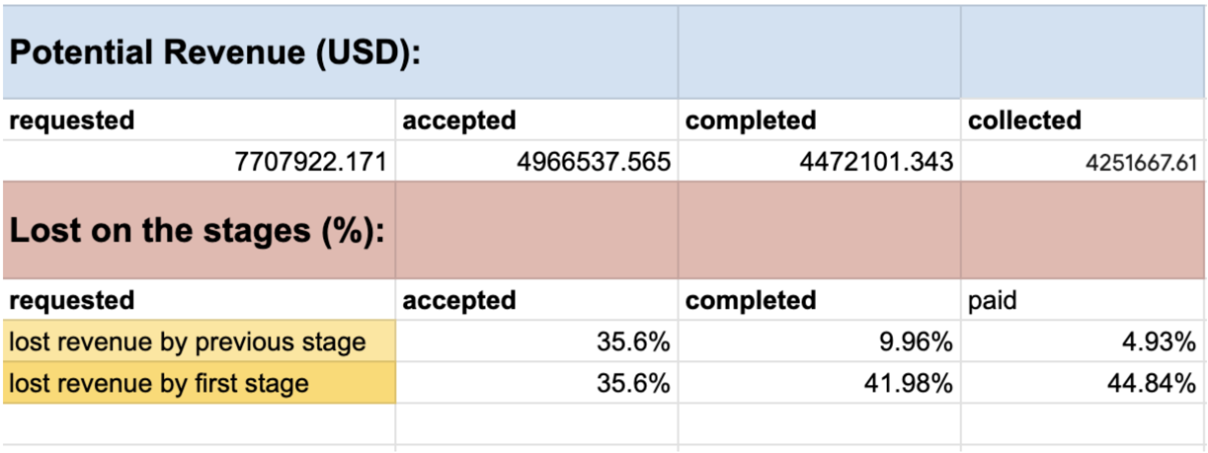
To assume which potential revenue could be received by improving conversion rates through funnel we moved to the ride granularity. Using tables transactions.csv and ride_requests.csv we calculated average amount USD per ride (Tab.1).



Total amount Approved			
4251667.61			
Number of payed rides			
212628			
Average amount by ride			
19.99580305			

Table 1.

Then we can compare potential revenue in every stage on ride granularity level (Tab.2). This calculation shows that we lost 35% of potential revenue when customers drop off after ride request and 42%% of revenue when we lost customers before ride completed.
Calculation has made in Google Sheets tool.



Potential Revenue (USD):			
requested	**accepted**	**completed**	**collected**
7707922.171	4966537.565	4472101.343	4251667.61
Lost on the stages (%):			
requested	**accepted**	**completed**	paid
lost revenue by previous stage	35.6%	9.96%	4.93%
lost revenue by first stage	35.6%	41.98%	44.84%

Table 2.

How to reduce your drop off rate

There are a number of reasons why people might drop off, such as website design issues. Reducing the drop off rate is essential for businesses because it can mean more sales and more customers. There are a number of ways to reduce the drop off rate:

- Conduct a survey with dropped off customers and drivers to define the causation of refusals
- Improving app and website design through A/B testing the app and website experience
- Reminding customers
- Implement a system of punishments and rewords of drivers

Appendix 1

```
WITH user_ride_status AS (
  SELECT user_id
  FROM ride_requests
  GROUP BY user_id
),
totals AS (
  SELECT
            platform,
           age_range,
            DATE(download_ts) AS download_dt,
    COUNT(DISTINCT a.*) AS total_users_downloads,
    COUNT(DISTINCT s.*) AS total_users_signed_up,
    COUNT(DISTINCT urs.user_id) AS total_users_ride_requested,
    COUNT(DISTINCT
      CASE WHEN accept_ts IS NOT NULL THEN rr.user_id END
    ) AS total_users_ride_accepted,
    COUNT(DISTINCT
      CASE WHEN dropoff_ts IS NOT NULL THEN rr.user_id END
    ) AS total_users_ride_completed,
    COUNT(DISTINCT
      CASE WHEN rr.ride_id IN (
        SELECT ride_id
        FROM transactions t
        WHERE charge_status = 'Approved'
      ) THEN rr.user_id END
    ) AS total_users_transactions,
    COUNT(DISTINCT r.user_id) AS total_users_reviews,

            COUNT(DISTINCT rr.ride_id) AS ride_count

  FROM app_downloads a
  LEFT JOIN signups s ON a.app_download_key = s.session_id
  LEFT JOIN user_ride_status urs ON urs.user_id = s.user_id
  LEFT JOIN ride_requests rr ON urs.user_id = rr.user_id
  LEFT JOIN transactions t ON rr.ride_id = t.ride_id
  LEFT JOIN reviews r ON rr.ride_id = r.ride_id
  GROUP BY platform, age_range, download_dt
),
funnel_stages AS (
  SELECT
    0 AS funnel_stage,
    'downloads' AS funnel_name,
            platform,
            age_range,
            download_dt,
    total_users_downloads AS user_count
  FROM totals

  UNION ALL

  SELECT
    1 AS funnel_stage,
            'signups' AS funnel_name,
            platform,
            age_range,
            download_dt,
            total_users_signed_up AS user_count
  FROM totals

  UNION ALL

  SELECT
    2 AS funnel_stage,
            'ride_request' AS funnel_name,
            platform,
            age_range,
            download_dt,
            total_users_ride_requested AS user_count
  FROM totals

  UNION ALL

  SELECT
    3 AS funnel_stage,
            'ride_accepted' AS funnel_name,
            platform,
            age_range,
            download_dt,
            total_users_ride_accepted AS user_count
  FROM totals
  UNION ALL

  SELECT
    4 AS funnel_stage,
            'ride_completed' AS funnel_name,
            platform,
             age_range,
            download_dt,
            total_users_ride_completed AS user_count
  FROM totals

  UNION ALL

  SELECT
    5 AS funnel_stage,
             'payment' AS funnel_name,
            platform,
            age_range,
            download_dt,
            total_users_transactions AS user_count
  FROM totals

  UNION ALL

  SELECT
    6 AS funnel_stage,
            'reviews' AS funnel_name,
             platform,
             age_range,
            download_dt,
            total_users_reviews AS user_count
  FROM totals
)
SELECT *
FROM funnel_stages
ORDER BY funnel_stage, platform, age_range;
```

→ funnel_aggregated_data.csv

Appendix 2

Ride Totals

SELECT		
COUNT(case when request_ts is not null		
THEN ride_id end) AS ride_requested,		
COUNT(case when accept_ts IS NOT NULL		
THEN ride_id end) AS ride_accepted,		
COUNT(CASE WHEN dropoff_ts IS NOT NULL		
then ride_id end) AS ride_completed		
FROM ride_requests		
ride_requested	ride_accepted	ride_completed

385477	248379	223652

Appendix 3

Tableau Public:
https://public.tableau.com/views/MetrocarFunnelAnalysis_16888434568450/MetrocarFunnelAnalysis?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link

Appendix 4

Google Sheets:
https://docs.google.com/spreadsheets/d/19PtcSfCVc1bFXLWjHnEB-n5gdmHhkhTQ_RD3x5AGKVQ/edit?usp=sharing