

# METROCAR FUNNEL ANALYSIS

By Tega odjugo

Performed on 2023-10-20

## Summary

An analysis was conducted on the Metrocar ride sharing app to identify areas of improvement and optimization. The aim of the analysis is to identify the steps users go through when they use the app and which steps need more focus. The result of the analysis shows that there is a significant decrease from when rides are accepted to when rides are completed.

**I recommend focusing on determining why riders are cancelling their ride before the driver arrives by sending surveys and reaching out to customers that cancel their ride. This will aid in better understanding rider's barriers and choose an appropriate strategy for addressing it.**

## Context

Metrocar's business model is based on a platform that connects riders with drivers through a mobile application. Metrocar acts as an intermediary between riders and drivers, providing a user-friendly platform to connect them and facilitate the ride-hailing process.

The customer funnel for Metrocar includes the following stages:

1. App Download: A user downloads the Metrocar app from the App Store or Google Play Store.
2. Signup: The user creates an account in the Metrocar app, including their name, email, phone number, and payment information.
3. Request Ride: The user opens the app and requests a ride by entering their pickup location, destination, and ride capacity (2 to 6 riders).
4. Driver Acceptance: A nearby driver receives the ride request and accepts the ride.
5. Ride: The driver arrives at the pickup location, and the user gets in the car and rides to their destination.
6. Payment: After the ride, the user is charged automatically through the app, and a receipt is sent to their email.
7. Review: The user is prompted to rate their driver and leave a review of their ride experience.

The metrocar data has 5 tables which are; app\_downloads, signups, ride\_requests, transactions and reviews.

## Results

The data extracted and analysed using SQL gives the following answers;

Downloads	There were 23,608 downloads from January 1 <sup>st</sup> 2021 to December 31 <sup>st</sup> 2021.
Signup	17,623 users signed up. 29.6% user drop off from signup to request.
Request ride	385,477 rides requested. 12,406 unique users requested a ride. Requests from different platforms are as follows; ios platform = 234,693, android platform = 112,317, web platform = 38,467.
Driver acceptance	248,379 rides were accepted by drivers.
Ride	223,652 rides were requested and completed. 52:36.738773 average time from pickup to drop-off.
Payment	212,628 rides had approved payment, 4,251,667.610000027 usd total paid.
Review	156,211 reviews were completed.

In order to answer the business questions raised, SQL queries were written that combine the data from different steps of the funnel and calculate the drop-off from one step to another. This enabled the following business questions to be answered:

- 1) What steps of the funnel should we research and improve? Are there any specific drop-off points preventing users from completing their first ride?

**Answer:** The steps of the funnel to be researched and improved are the driver accepted to the ride completed steps, which shows that 50.77% of rides accepted by drivers are actually completed. This drop off point is preventing users from completing their first ride. The reviews completed step should also be researched as it shows that 69.76% of users that complete their payment actually send in a review.

- 2) Metrocar currently supports 3 different platforms: ios, android, and web. To recommend where to focus our marketing budget for the upcoming year, what insights can we make based on the platform?

**Answer:** The platform that supports the most request is the IOS platform which hosted 60.8% of total rides requested from all platforms. This shows that more users are aligned

with using IOS for taking their trips. The marketing budget will yield more result from focusing on this market.

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

**Answer:** After cleaning up the unknown and null age groups, the age group that performs the best at each stage of the funnel is the 35-44 age group. This is the age group of our target customers. The next age group according to performance is the 25-34 age group.

- 4) Surge pricing is the practice of increasing the price of goods or services when there is the greatest demand for them. If we want to adopt a price-surfing strategy, what does the distribution of ride requests look like throughout the day?

**Answer:** The hours of the day that has the highest ride volume is 8-10am and 4-8pm. We can conclude from this that workers with standard daytime hours are using Metrocar to commute.

- 5) What part of our funnel has the lowest conversion rate? What can we do to improve this part of the funnel?

**Answer:** The ride completed funnel has the lowest conversion rate. To improve this part of the funnel we can send out surveys and reach out to customers that cancelled their ride in order to determine the reasons they cancelled. A strategy can be developed to address the reoccurring reasons in order to reduce ride cancellation.

Tableau is used to visualize data set extracted from SQL and the following visuals are the result

## Percent of Top Vs Percent of Previous

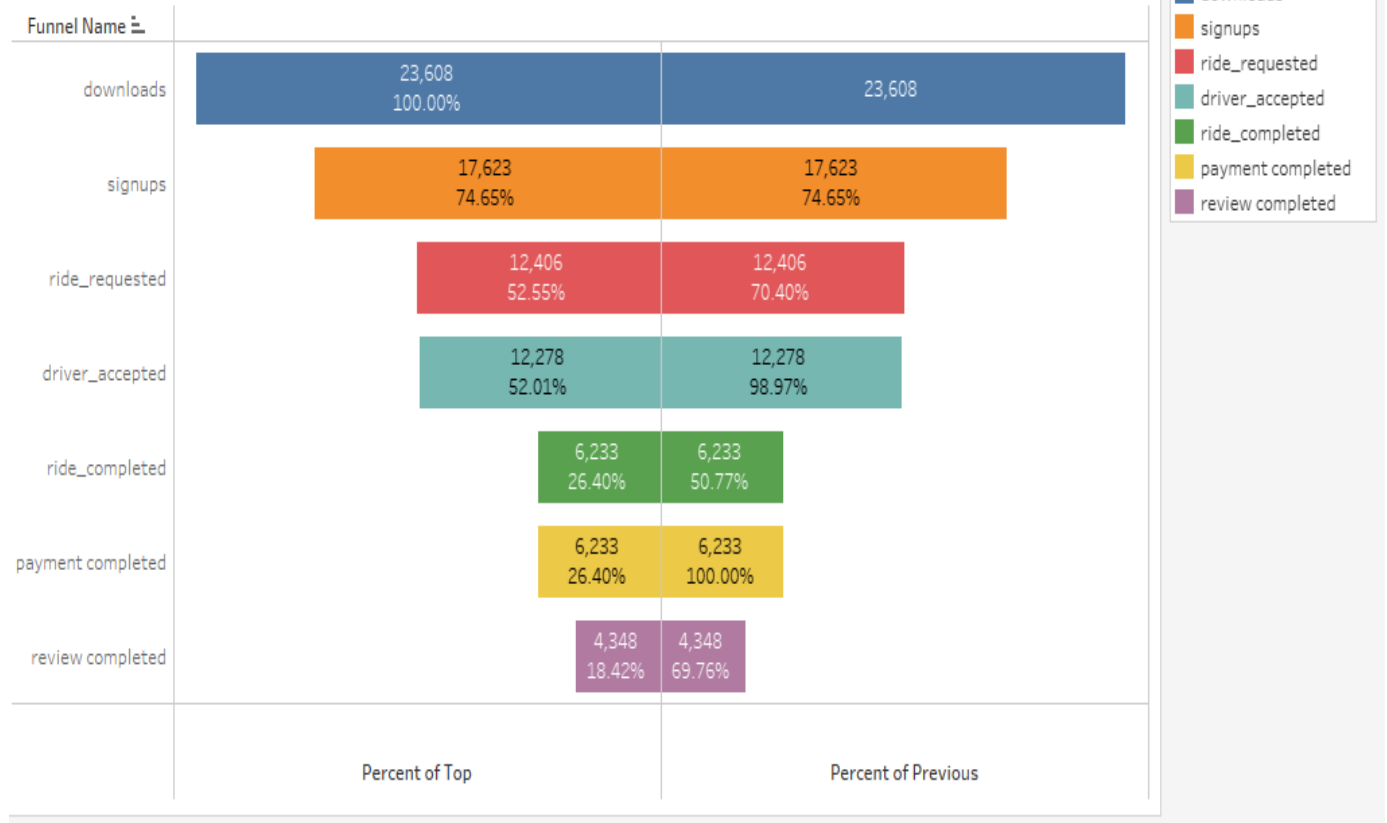


Fig 1 Percent of Top vs Percent of previous for user count at each funnel stage

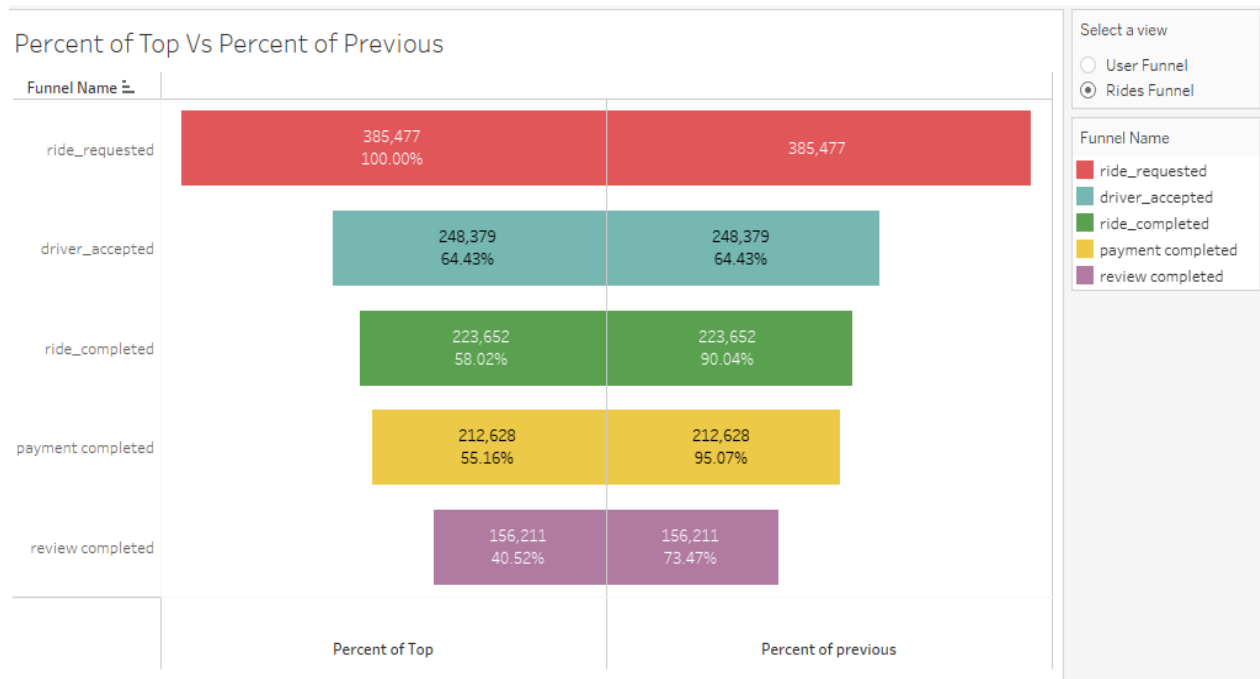


Fig 2 Percent of Top vs Percent of previous for rides count at each funnel stage

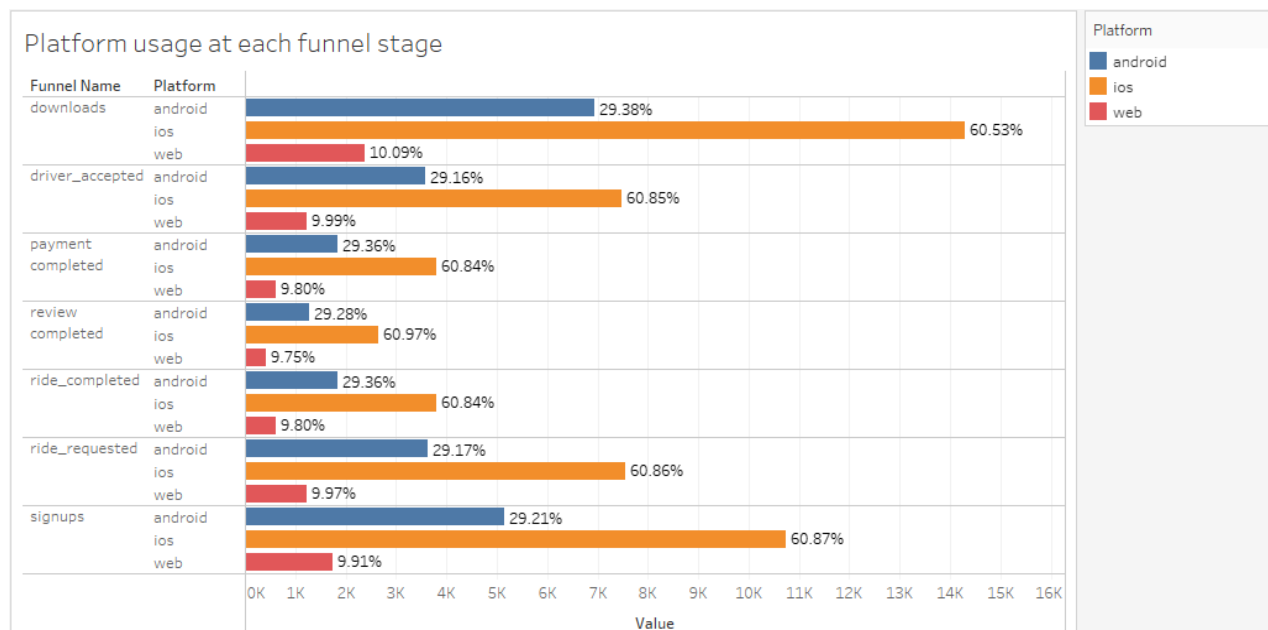


Fig 3 Platform usage at each funnel step

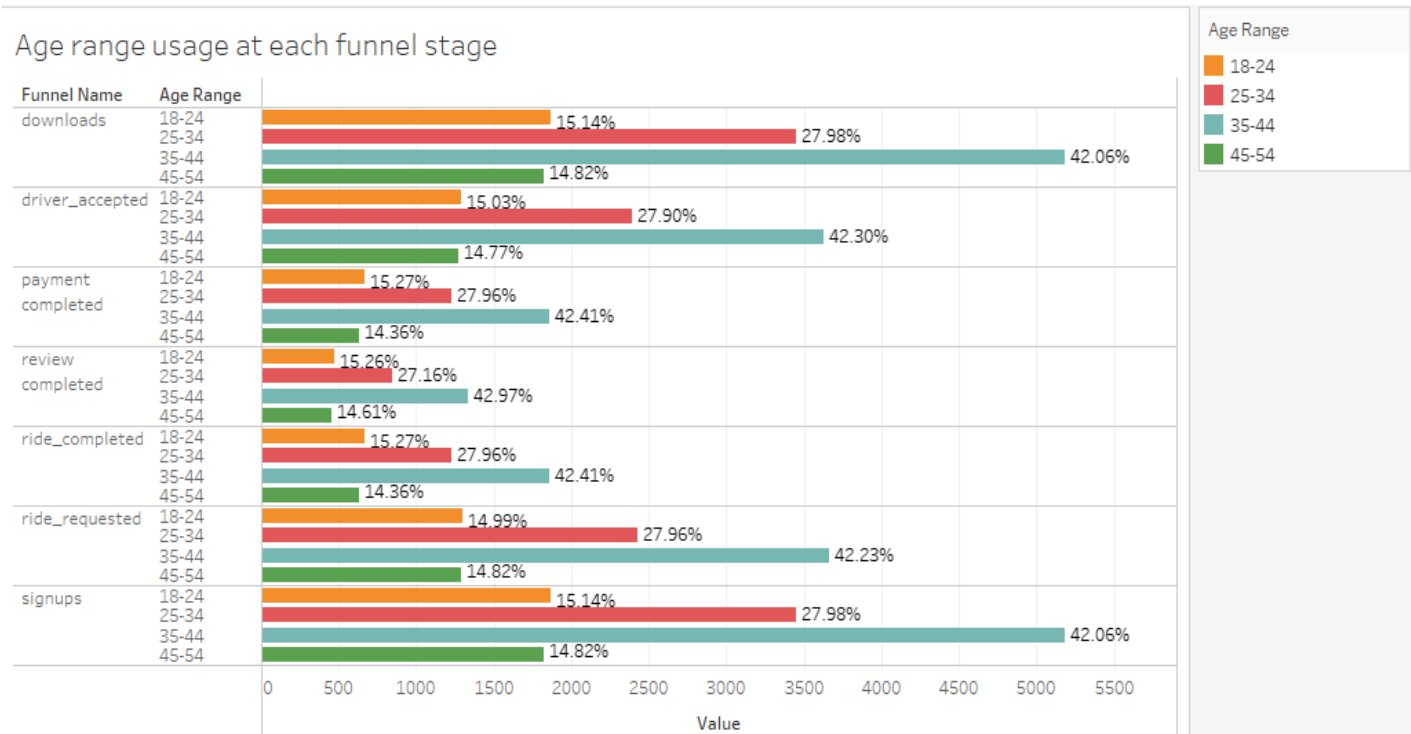


Fig 4 Age range usage at each funnel step

## Recommendation

I recommend focusing on determining why riders are cancelling their ride before the driver arrives by sending surveys and reaching out to customers that cancel their ride. This will aid in better understanding rider's barriers and choose an appropriate strategy for addressing it.

# Appendix

## 1) SQL code for unique users that requested a ride

```
WITH user_ride_status
  AS (SELECT user_id,
            Max(CASE
                WHEN dropoff_ts IS NOT NULL THEN 1
                ELSE 0
            END) AS ride_completed
  FROM   ride_requests
  GROUP BY user_id)
SELECT Count(*) AS total_users_ride_requested
FROM   user_ride_status;
```

## 2) SQL code for total ride requested and completed

```
WITH user_ride_status
  AS (SELECT ride_id,
            Max(CASE
                WHEN dropoff_ts IS NOT NULL THEN 1
                ELSE 0
            END) AS ride_completed
  FROM   ride_requests
  GROUP BY ride_id)
SELECT Sum(ride_completed) AS total_users_ride_completed
FROM   user_ride_status;
```

## 3) SQL code for aggregated funnel analysis

```
WITH user_downloads
  AS (SELECT Count(app_download_key) AS total_download,
            platform,
            age_range,
            Date_trunc('day', download_ts) AS download_dt
  FROM   app_downloads
  LEFT JOIN signups
        ON app_downloads.app_download_key = signups.session_id
  GROUP BY platform,
            age_range,
            Date_trunc('day', download_ts)),
driver_acceptance
  AS (SELECT Count(DISTINCT CASE
                        WHEN accept_ts IS NOT NULL THEN
                            ride_requests.user_id
                        END) AS rides_accepted_by_driver,
            Count (accept_ts) AS total_accepted,
            platform,
            age_range,
            Date_trunc('day', download_ts) AS download_dt
  FROM   ride_requests
  LEFT JOIN signups
```

```

        ON ride_requests.user_id = signups.user_id
    LEFT JOIN app_downloads
        ON signups.session_id = app_downloads.app_download_key
    GROUP BY platform,
            age_range,
            Date_trunc('day', download_ts)),
ride_completed
AS (SELECT Count(DISTINCT CASE
                        WHEN dropoff_ts IS NOT NULL THEN
                            ride_requests.user_id
                        END)
    Count(dropoff_ts)
    AS users_completing_a_ride,
    AS total_rides,
    platform,
    age_range,
    Date_trunc('day', download_ts) AS download_dt
FROM ride_requests
LEFT JOIN signups
    ON ride_requests.user_id = signups.user_id
LEFT JOIN app_downloads
    ON signups.session_id = app_downloads.app_download_key
GROUP BY platform,
        age_range,
        Date_trunc('day', download_ts)),
user_ride_status
AS (SELECT user_id
FROM ride_requests),
totals
AS (SELECT Count(DISTINCT s.user_id)
    Count(DISTINCT urs.user_id)
    Count(urs.user_id)
    AS total_users_signed_up,
    AS total_users_ride_requested,
    AS ride_requested,
    platform,
    age_range,
    Date_trunc('day', download_ts) AS download_dt
FROM signups s
LEFT JOIN user_ride_status urs
    ON s.user_id = urs.user_id
LEFT JOIN app_downloads
    ON s.session_id = app_downloads.app_download_key
GROUP BY platform,
        age_range,
        Date_trunc('day', download_ts)),
payment_completed
AS (SELECT Count(DISTINCT CASE
                        WHEN rs.dropoff_ts IS NOT NULL THEN
                            rs.user_id
                        END)
    Count(dropoff_ts)
    AS rides_paid_for,
    AS total_payment,
    platform,
    age_range,
    Date_trunc('day', download_ts) AS download_dt
FROM ride_requests rs
LEFT JOIN transactions ts
    ON rs.ride_id = ts.ride_id
LEFT JOIN signups
    ON rs.user_id = signups.user_id
LEFT JOIN app_downloads
    ON signups.session_id = app_downloads.app_download_key

```



```

GROUP BY platform,
        ts.charge_status,
        age_range,
        Date_trunc('day', download_ts)
HAVING ts.charge_status LIKE 'Approved%'),
review_completed
AS (SELECT Count(DISTINCT reviews.user_id) AS review_count,
        Count (reviews.user_id)          AS total_reviews,
        platform,
        age_range,
        Date_trunc('day', download_ts)   AS download_dt
FROM reviews
LEFT JOIN signups
        ON reviews.user_id = signups.user_id
LEFT JOIN app_downloads
        ON signups.session_id = app_downloads.app_download_key
GROUP BY platform,
        age_range,
        Date_trunc('day', download_ts)),
funnel_stages
AS (SELECT 1          AS funnel_step,
        'downloads'   AS funnel_name,
        total_download AS value,
        0             AS ride_count,
        platform,
        age_range,
        download_dt
FROM user_downloads
UNION
SELECT 2          AS funnel_step,
        'signups'     AS funnel_name,
        total_users_signed_up AS value,
        0           AS ride_count,
        platform,
        age_range,
        download_dt
FROM totals
UNION
SELECT 3          AS funnel_step,
        'ride_requested' AS funnel_name,
        total_users_ride_requested AS value,
        ride_requested AS ride_count,
        platform,
        age_range,
        download_dt
FROM totals
UNION
SELECT 4          AS funnel_step,
        'driver_accepted' AS funnel_name,
        rides_accepted_by_driver AS value,
        total_accepted AS ride_count,
        platform,
        age_range,
        download_dt
FROM driver_acceptance
UNION
SELECT 5          AS funnel_step,

```

```

        'ride_completed' AS funnel_name,
        users_completing_a_ride AS value,
        total_rides AS ride_count,
        platform,
        age_range,
        download_dt
FROM ride_completed
UNION
SELECT 6 AS funnel_step,
        'payment completed' AS funnel_name,
        rides_paid_for AS value,
        total_payment AS ride_count,
        platform,
        age_range,
        download_dt
FROM payment_completed
UNION
SELECT 7 AS funnel_step,
        'review completed' AS funnel_name,
        review_count AS value,
        total_reviews AS ride_count,
        platform,
        age_range,
        download_dt
FROM review_completed)
SELECT *
FROM funnel_stages
ORDER BY funnel_step;

```

4) Tableau public link: [Metrodashboard](#)