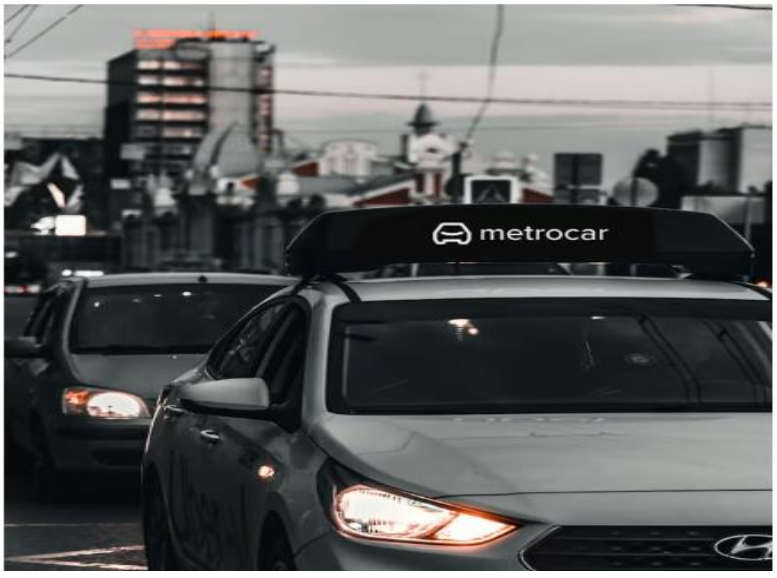
Vanessa Kimana
2023-11-22

Metrocar
Customer Funnel Analysis



Vanessa Kimana
2023-11-22

1. Summary

This project aims to analyze the customer funnel of Metrocar, a ride-sharing app. It offers a comprehensive analysis of Metrocar's customer funnel, aiming to identify areas for improvement and optimization. Furthermore, this report aims to address the aspects of the funnel that should be researched and improved and to identify any specific drop-off points that might be preventing users from completing their ride. Metrocar currently supports three different platforms: iOS, Android, and web. Insights based on each platform will guide recommendations on where to focus the marketing budget for the upcoming year. The report seeks to determine which age groups excel at each stage of the funnel and represent our target customers. Analyzing the daily distribution of ride requests will inform the adoption of a surge pricing strategy. Additionally, the report aims to identify the funnel segment with the lowest conversion rate and propose strategies for improvement.

1. Project Background

Metrocar operates on a business model centered around a platform that links riders and drivers via a mobile application. Serving as an intermediary, Metrocar offers a user-friendly platform to connect and facilitate the ride-hailing process for both riders and drivers.

2. Context and Metrocar’s Customer Funnel

The customer funnel for Metrocar typically includes the following steps:

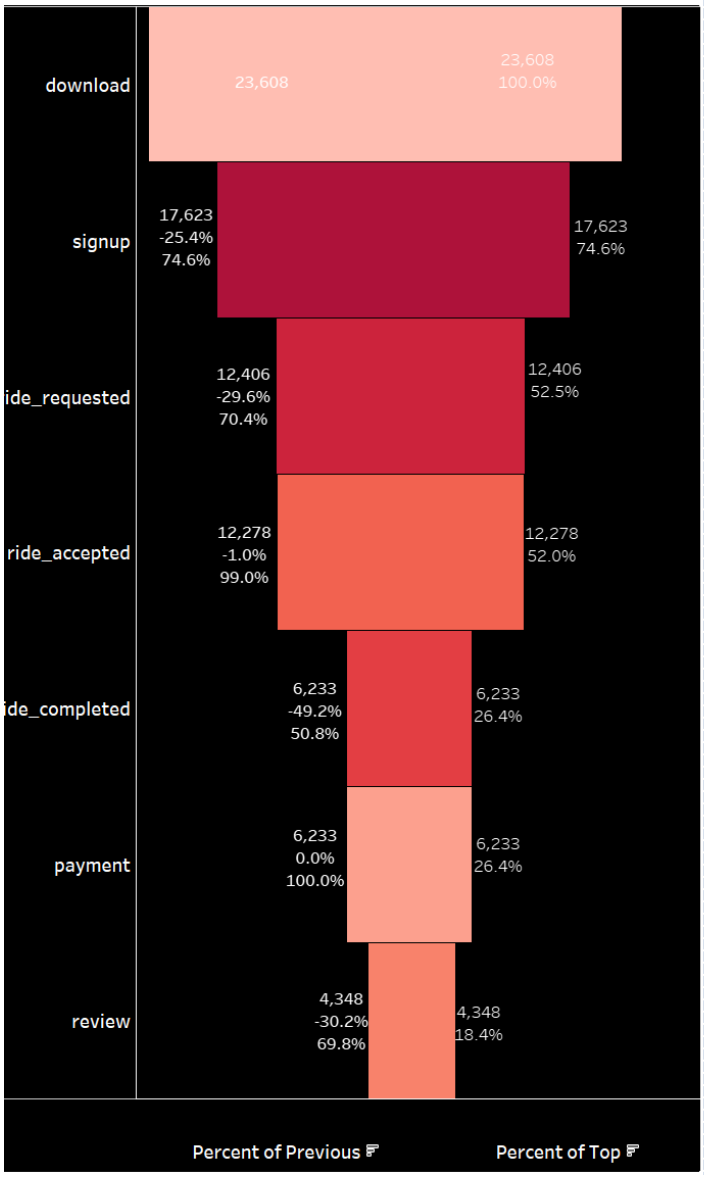
1. App Download: A user downloads the Metrocar app from the App Store or Google Play Store.
2. Signup: The user creates an account in the Metrocar app, including their name, email, phone number, and payment information.
3. Request Ride: The user opens the app and requests a ride by entering their pickup 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

3. Data Context and results

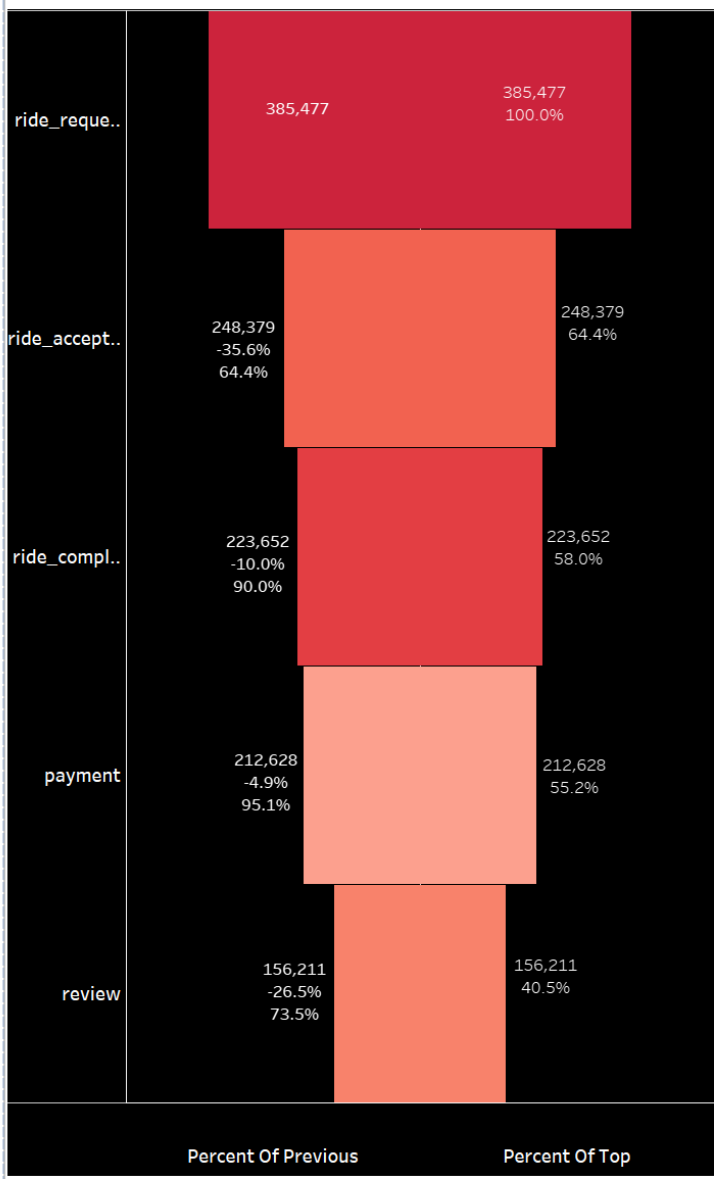
The data below illustrates the user count and ride count of the funnel where the right side of the funnel presents the total count of user and ride, followed by the drop-off rate and the conversation rates. It also calculates the percentage of previous which indicates the percentage of users/rides at each stage relative to the previous stage. The left side of the funnel presents the sum of total user count and ride along with the conversion rates. It further calculates the percentage of top, which indicates the percentage of users/rides at each stage of the funnel relative to the top of the funnel.

Vanessa Kimana
2023-11-22

User Funnel



Ride Funnel



Vanessa Kimana
2023-11-22

Observations:

There is a notable decline in user numbers from the "Download" to the "Signup" stage, suggesting potential obstacles in continuing the process.
There is a significant drop-off between "Ride Accepted" and "Ride Completed." This could be explored by investigating potential user experience issues, delays, or obstacles in the completion process.
The review stage shows a significant drop-off, of 30%.

In the ride funnel:

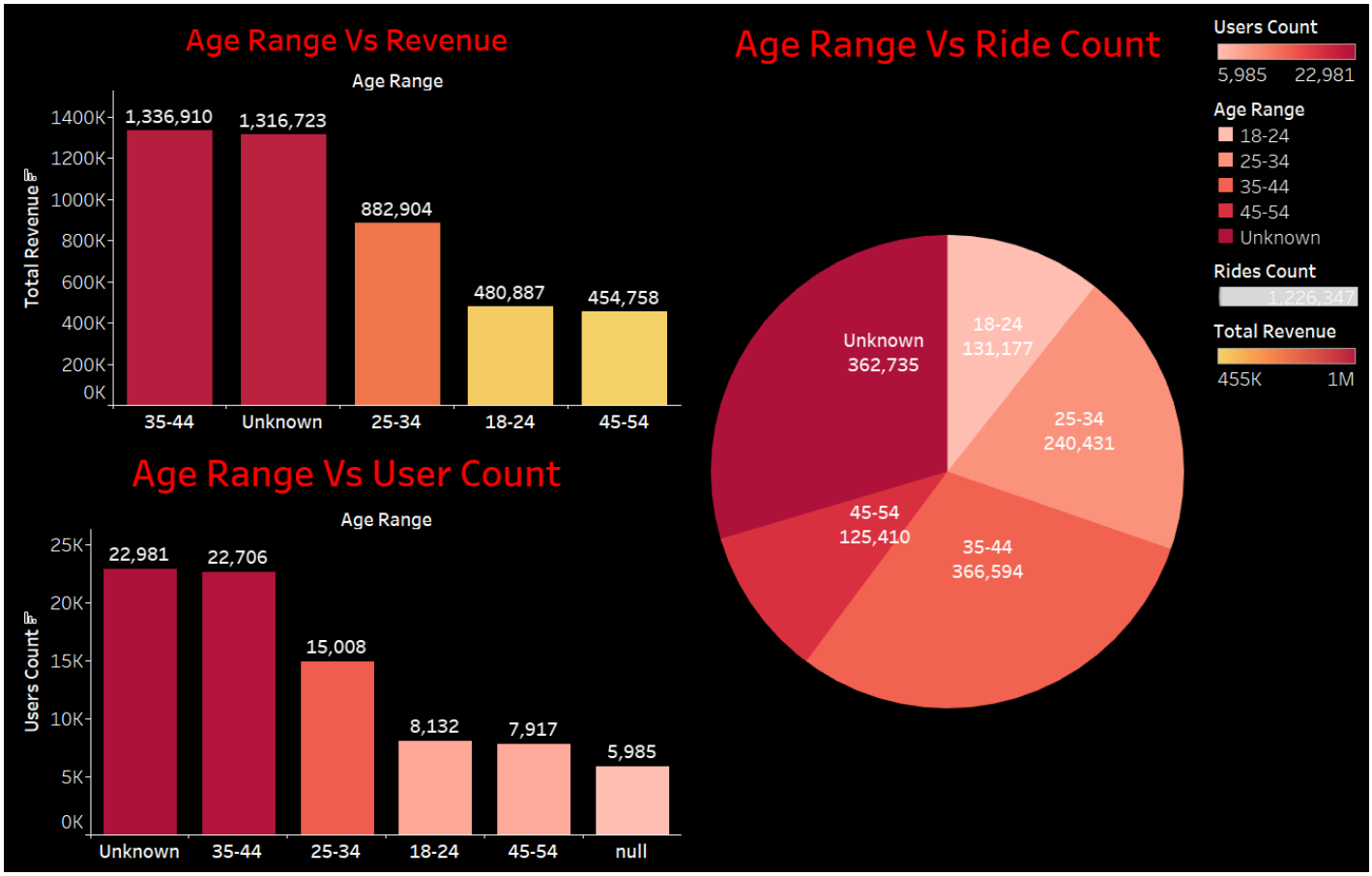
Substantial drop in users from "Download" to "Signup" of 25%
Significant drop-off between "Ride Accepted" and "Ride Completed.“ of 29%
Noticeable drop-off at the "Review" stage of 30%
While the overall conversion rate is relatively high, it's essential to understand the reasons behind the drop-offs.

Age range Analysis:

In the revenue breakdown, the category labeled as "Unknown" emerges as the top contributor, signifying the highest revenue. Notably, the age range of 35-44 stands out prominently as a major revenue generator for Metrocar. Meanwhile, the youngest users, falling within the 18-24 age range, contribute significantly to revenue, amounting to $480,887.

When considering user count, the largest user base is observed in the 35-44 age range, followed by the 25-34 age group and then the 18-24 age group. It's worth mentioning that the "Unknown" category also boasts a substantial user count, adding to the overall diversity and reach of Metrocar's user demographics. See figure below for Age range Analysis:

Vanessa Kimana
2023-11-22

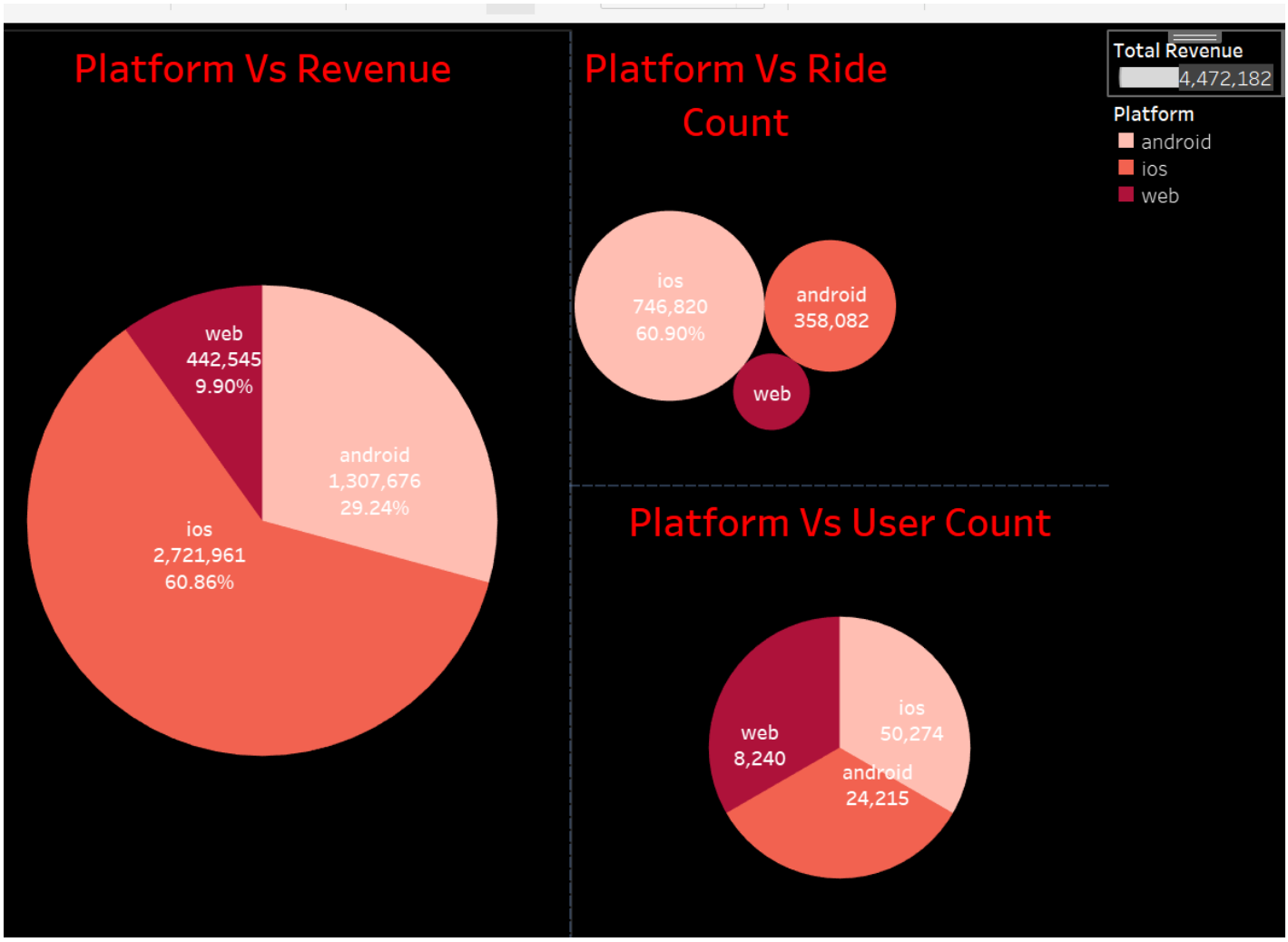


Platform Analysis:

In both revenue generation and user count, IOS takes the lead, constituting 60.86% of the total user base for Metrocar. Following closely, Android stands as the second-largest platform, contributing significantly to both user count and revenue.

Conversely, the web platform, while the smallest among the three, still makes a notable contribution, accounting for 9.90% of the total. This emphasizes the importance of considering all platforms in Metrocar's marketing and service strategies. See figure below for platform Analysis:

Vanessa Kimana
2023-11-22

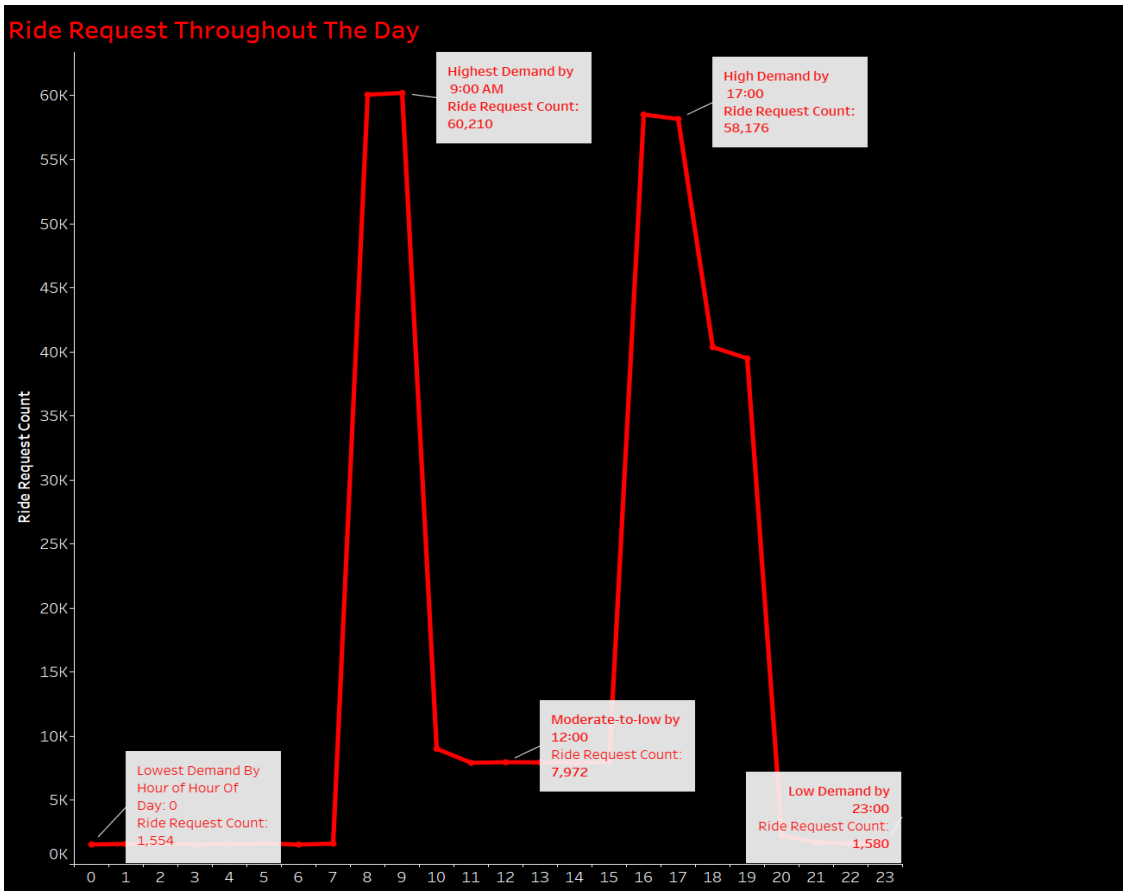


Ride request throughout the day:

There is a substantial demand for ride requests during rush hours, particularly during the peak hours of 8:00 AM - 9:00 AM and 5:00 PM - 6:00 PM. Additionally, there is notable activity during the late-night hours, suggesting a moderate demand for rides during that time.

Conversely, the hours between 00:00 and 05:00 exhibit the lowest ride requests, indicating off-peak hours with reduced demand for Metrocar services. See figure below for hours analysis Analysis:

Vanessa Kimana
2023-11-22



4. Areas of Optimization and improvement

Based on these findings it is recommended to:

- Optimize the user funnel by improving the signup process to address the 25.4% drop-off prior download step, enhancing ride request engagement to focus on the 29.6% drop-off and optimize the review process to encourage more users to provide feedback.
- Optimize the ride funnel by addressing the 35.6% drop-off between ride request and acceptance, explore user friendly payment step by minimizing the 4.9% drop-off and encouraging more riders to leave feedback.
- Implementing strategies to capture the demographics within this “unkown” category. Intensify marketing for the lucrative 35-44 age group to increase engagement.
- Focus on iOS for revenue optimization (60.86%) and enhance Android engagement (29.24%).
- Implement surge pricing during peak hours, particularly during morning and evening rush hours, to maximize revenue during high-demand periods. Enhance dynamic pricing adjustments during late-night hours to optimize earnings during periods of off peak.

Vanessa Kimana
2023-11-22

Appendix:

```
--How many times was the app downloaded?
SELECT COUNT(*) FROM app_downloads;

SELECT platform, COUNT(*)
FROM app_downloads
GROUP BY platform;

--How many users signed up on the app?
SELECT COUNT(DISTINCT(user_id)) AS tot_users_signups
FROM signups;
--------------------------------------------------------------------------------
--How many rides were requested through the app?
SELECT COUNT(ride_id) AS tot_requested_rides
FROM ride_requests;

--------------------------------------------------------------------------------
--How many rides were requested and completed through the app?
SELECT
SUM(CASE
WHEN dropoff_ts IS NOT NULL THEN 1 ELSE 0 END) AS completed_rides,
COUNT(ride_id) AS tot_requested_rides
FROM ride_requests;

--------------------------------------------------------------------------------
--How many rides were requested and how many unique users requested a ride?
-- Revise to learn subquery
SELECT
COUNT(DISTINCT user_id) AS  unique_users_requested_ride,
(SELECT COUNT(ride_id)) AS tot_requested_rides
FROM ride_requests;

WITH rider_counts AS
(SELECT
COUNT(DISTINCT user_id) AS  unique_users_requested_ride,
(SELECT COUNT(ride_id)) AS tot_requested_rides
FROM ride_requests)
SELECT unique_users_requested_ride, tot_requested_rides
FROM rider_counts;

SELECT
COUNT(DISTINCT user_id) AS  unique_users_requested_ride,
COUNT(ride_id) AS tot_requested_rides
FROM ride_requests;
--------------------------------------------------------------------------------
--What is the average time of a ride from pick up to drop off?

SELECT AVG(dropoff_ts - pickup_ts) AS average_time_dropoff_pickup
FROM ride_requests
WHERE dropoff_ts IS NOT NULL AND pickup_ts IS NOT NULL;

----------------------------------------------------------------------------------------------------
--How many rides were accepted by a driver?
SELECT COUNT(accept_ts) AS accepted_driver FROM ride_requests
WHERE accept_ts IS NOT NULL;

----------------------------------------------------------------------------------------------------
--How many rides did we successfully collect payments and how much was collected?
SELECT COUNT (ride_id)as tot_ride, SUM(purchase_amount_usd) AS uccessfully_payments
```

Vanessa Kimana
2023-11-22

```
FROM transactions
WHERE charge_status = 'Approved';
-----------------------------------------------------------------------------------------------------
--How many ride requests happened on each platform?
SELECT platform, COUNT(ride_id) AS tot_ride_requests_platform
FROM metrocar
GROUP BY platform;
-----------------------------------------------------------------------------------------------------
--What is the drop-off rate from users signing up to users requesting a ride
SELECT tot_signups,
      tot_requested_ride,
      (tot_requested_ride * 100.0 /tot_signups) AS dropoff_rate
 FROM
(SELECT COUNT(DISTINCT user_id) AS tot_signups
FROM signups) AS s,
(SELECT COUNT(DISTINCT user_id) AS tot_requested_ride
FROM ride_requests) AS rr

-----------------------------------------------------------------------------------------------
SELECT
   COUNT(DISTINCT user_id) AS users_requesting_a_ride,
   COUNT(DISTINCT
      CASE
         WHEN dropoff_ts IS NOT NULL
         THEN user_id
      END
   ) AS users_completing_a_ride
FROM ride_requests;
-----------------------------------------------------------------------------------------------
SELECT
   *
   FROM metrocar;
-----------------------------------------------------------------------------------------------
WITH user_ride_status AS (
   SELECT
      user_id
   FROM ride_requests
   GROUP BY user_id
)
SELECT
   COUNT(*) AS total_users_signed_up,
   COUNT(DISTINCT urs.user_id) AS total_users_ride_requested
FROM signups s
LEFT JOIN user_ride_status urs ON
   s.user_id = urs.user_id;
-----------------------------------------------------------------------------------------------
SELECT app_download_key,
      user_id, platform,
      age_range,
   DATE(download_ts) AS download_dt FROM app_downloads
 LEFT JOIN signups ON app_downloads.app_download_key = signups.session_id
-----------------------------------------------------------------------------------------------
-----------------------------------------------------------------------------------------------
SELECT
   LPAD(EXTRACT(HOUR FROM request_ts)::TEXT, 2, '0') || ':00:00' AS hour_of_day,
   COUNT(*) AS ride_request_count
FROM
   ride_requests
GROUP BY
   hour_of_day
ORDER BY
```

Vanessa Kimana
2023-11-22

```
   hour_of_day;

-------------------------------------------------------------------------------
-- Step 1: Extract user details from app_downloads and signups
WITH user_details AS (
  SELECT
    ad.app_download_key,
    su.user_id,
    ad.platform,
    su.age_range,
    DATE(ad.download_ts) AS download_dt
  FROM
    app_downloads ad
  LEFT JOIN
    signups su ON ad.app_download_key = su.session_id
),

-- Step 2: Define the funnel steps and calculate counts
funnel_data AS (
  --Download step
  SELECT
    0 AS step_count,
    'download' AS name,
    platform,
    age_range,
    download_dt,
    COUNT(DISTINCT app_download_key) AS users_count,
    0 AS rides_count
  FROM
    user_details
  GROUP BY
    platform, age_range, download_dt

  UNION ALL

  --Signup step
  SELECT
    1 AS step_count,
    'signup' AS name,
    platform,
    age_range,
    download_dt,
    COUNT(DISTINCT user_id) AS users_count,
    0 AS count_rides
  FROM
    user_details
  GROUP BY
    platform, age_range, download_dt

  UNION ALL

  --Ride Requested step
  SELECT
    2 AS step_count,
    'ride_requested' AS name,
    platform,
    age_range,
    download_dt,
    COUNT(DISTINCT user_id) AS users_count,
    COUNT(DISTINCT ride_id) AS rides_count
  FROM
```

```
  ride_requests
JOIN
  user_details USING (user_id)
WHERE
  request_ts IS NOT NULL
GROUP BY
  platform, age_range, download_dt

UNION ALL

--Ride Accepted step
SELECT
  3 AS step_count,
  'ride_accepted' AS name,
  platform,
  age_range,
  download_dt,
  COUNT(DISTINCT user_id) AS users_count,
  COUNT(DISTINCT ride_id) AS rides_count
FROM
  ride_requests
JOIN
  user_details USING (user_id)
WHERE
  accept_ts IS NOT NULL
GROUP BY
  platform, age_range, download_dt

UNION ALL

--Ride Completed step
SELECT
  4 AS step_count,
  'ride_completed' AS name,
  platform,
  age_range,
  download_dt,
  COUNT(DISTINCT user_id) AS users_count,
  COUNT(DISTINCT ride_id) AS rides_count
FROM
  ride_requests
JOIN
  user_details USING (user_id)
WHERE
  cancel_ts IS NULL
GROUP BY
  platform, age_range, download_dt

UNION ALL

--Payment step
SELECT
  5 AS step_count,
  'payment' AS name,
  platform,
  age_range,
  download_dt,
  COUNT(DISTINCT user_id) AS users_count,
  COUNT(DISTINCT ride_id) AS rides_count
FROM
  transactions
```

```
  JOIN
    ride_requests USING (ride_id)
  JOIN
    user_details USING (user_id)
  WHERE
    charge_status = 'Approved'
  GROUP BY
    platform, age_range, download_dt

  UNION ALL

  --Review step
  SELECT
    6 AS step_count,
    'review' AS name,
    platform,
    age_range,
    download_dt,
    COUNT(DISTINCT user_id) AS users_count,
    COUNT(DISTINCT ride_id) AS rides_count
  FROM
    reviews
  JOIN
    user_details USING (user_id)
  GROUP BY
    platform, age_range, download_dt
)

-- Step 3: Display results
SELECT * FROM funnel_data
ORDER BY step_count, platform, age_range, download_dt;

-----------------------------------------------------------------------------------------------------------
WITH user_details AS (
  SELECT
    s.age_range,
    ad.platform,
    t.purchase_amount_usd
  FROM
    signups s
  LEFT JOIN
    ride_requests r ON s.user_id = r.user_id
  LEFT JOIN
    transactions t ON r.ride_id = t.ride_id
  LEFT JOIN
    app_downloads ad ON s.session_id = ad.app_download_key
)

SELECT
  age_range,
  platform,
  SUM(purchase_amount_usd) AS total_revenue
FROM
  user_details
GROUP BY
  age_range, platform
ORDER BY
platform, age_range;
```

Vanessa Kimana
2023-11-22

https://public.tableau.com/app/profile/vanessa.kimana/viz/MetrocarAnalysis_17008012557300/MetrocarFunnelAnalysis?publish=yes

Excel Zipped file.

Vanessa Kimana
2023-11-22