10/8/2023

Funnel Analysis

Metrocar
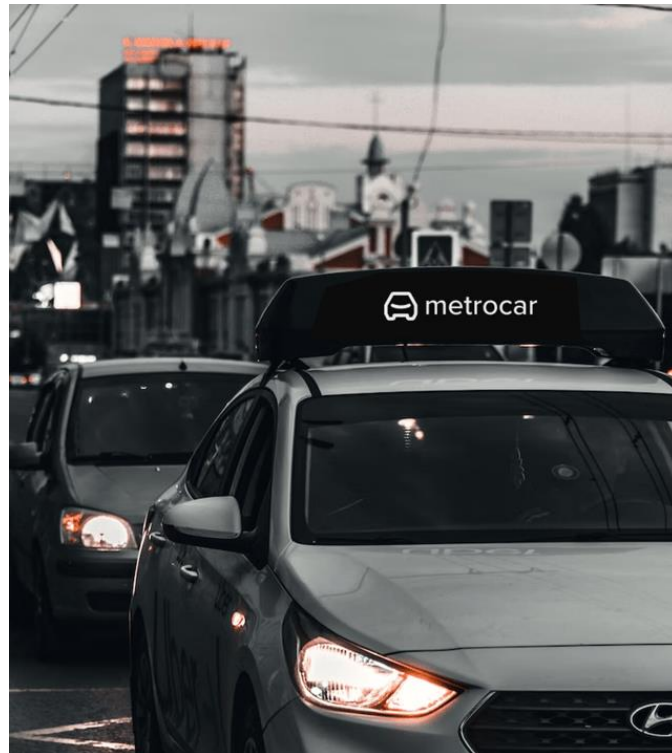
Robert Tiger
MASTERSCHOOL

Robert Tiger

Contents

Robert Tiger

Robert Tiger

Background

This project aims to analyze the customer funnel of Metrocar, a ride-sharing app (similar to Uber/Lyft), to identify areas for improvement and optimization. We used SQL to query the data and Tableau or Google Sheets for data visualization.

We have been asked by the stakeholders several business questions that uncovered valuable insights for improving specific areas of the customer funnel. We were tasked to conduct a funnel analysis and address the business questions. Furthermore, we will explain our reasoning behind the recommendations based on insights retrieved from the data.

Robert Tiger

Prologue: Funnel Analysis Basics

Funnel Analysis in the Ride-Sharing Industry

Funnel analysis is a widely employed methodology across various industries, and it holds particular significance in the realm of ride-sharing services. Ride-sharing platforms, like Metrocar, rely heavily on user engagement and conversion through their customer funnel. In this section, we will delve into the relevance and impact of funnel analysis within the ride-sharing industry.

Understanding the Ride-Sharing Funnel:

The ride-sharing funnel is a series of sequential stages that users go through, from their initial interaction with the app to completing a ride and providing feedback. Each stage presents an opportunity for the platform to engage and retain users, as well as generate revenue. Below, we outline the primary stages of the ride-sharing funnel:

1. **App Download:** Users discover the ride-sharing service, typically through app stores like the App Store or Google Play. They decide to download the app based on factors like brand reputation and marketing efforts.
2. **Signup:** After downloading the app, users must complete the signup process. This involves creating an account, providing personal information, and linking payment methods.
3. **Request Ride:** Users open the app and request a ride by specifying their pickup location, destination, and ride preferences.
4. **Driver Acceptance:** Upon receiving a ride request, a nearby driver decides whether to accept it or not. This stage determines how quickly a ride can be secured.
5. **Rides:** The driver transports the user to their destination. The ride experience directly impacts user satisfaction and retention.
6. **Payment:** After the ride is completed, the app automatically charges the user for the service. Payment processing is a critical part of the funnel, as it directly affects revenue.
7. **Review:** Users are prompted to rate their driver and leave feedback about their ride experience. Reviews help maintain service quality and build trust.

The Importance of Funnel Analysis in Ride-Sharing

Funnel analysis serves several crucial purposes within the ride-sharing industry:

- **Identifying Drop-Off Points:** By tracking user behavior at each stage of the funnel, ride-sharing platforms can pinpoint where users drop off or abandon the process. This identification is pivotal for understanding user pain points and opportunities for improvement.

- **Optimizing User Experience:** Insights gained from funnel analysis enable ride-sharing services to enhance the user experience at specific funnel stages. This might involve streamlining signup processes, reducing wait times for ride requests, or addressing payment issues.

- **Increasing Conversion Rates:** Improvements based on funnel analysis can lead to higher conversion rates, meaning more users complete the desired actions, such as registering, requesting rides, and providing reviews.

- **Revenue Generation:** Enhancing the funnel can have a direct impact on revenue. Higher conversion rates, improved payment processing, and user satisfaction can all contribute to increased earnings.

Robert Tiger

Industry Best Practice

In the competitive landscape of ride-sharing, staying ahead requires continuous optimization and adaptation. Ride-sharing companies often employ A/B testing, data-driven decision-making, and user feedback analysis to fine-tune their funnels and remain competitive.

Funnel analysis serves as the cornerstone of these practices, providing the data-driven insights necessary for informed decision-making and strategic planning.

Robert Tiger

Prologue: Metrocar's Customer Funnel

In this chapter, we take a closer look at Metrocar's customer funnel, which consists of several key stages that users go through when using the ride-sharing app. Understanding these stages is essential for conducting a comprehensive funnel analysis and identifying areas for improvement.

Overview of Metrocar's Funnel Stages

Metrocar's customer funnel encompasses a series of sequential stages that users navigate, starting from their initial interaction with the app to completing a ride and providing feedback. Each stage plays a vital role in the overall user experience and contributes to the success of the platform. Below, we provide an overview of Metrocar's funnel stages:

1. **App Download:** The journey begins when a user discovers Metrocar and decides to download the app from either the App Store or Google Play Store. This is the first step towards becoming a Metrocar user.

2. **Signup:** After downloading the app, users are required to complete the signup process. During signup, users provide essential information, including their name, email address, phone number, and payment details. A successful signup results in the creation of a user account.

3. **Request Ride:** Once registered, users can open the Metrocar app and request a ride by specifying their pickup location, destination, and ride capacity preferences (e.g., 2 to 6 riders). This stage marks the initiation of a ride request.

4. **Rides Accepted by Driver:** After a user submits a ride request, it is sent to nearby drivers, who have the option to accept or decline the ride. The acceptance of a ride request by a driver is a critical step in the process, as it determines whether the user can proceed with the ride.

5. **Customer Pick-Ups:** Once a driver accepts a ride request, they physically arrive at the user's specified pickup location. At this point, the user enters the driver's vehicle, marking the beginning of the ride.

6. **Customers Dropped-Off:** The driver transports the user to their specified destination. This stage represents the core service provided by Metrocar, as users rely on the platform to reach their intended locations safely and efficiently.

7. **Payment Completed:** Upon completing the ride, Metrocar automatically processes payment through the app. This seamless payment system ensures a hassle-free experience for users, with charges being applied to the user's preferred payment method.

8. **Customers Rated:** After the ride, Metrocar prompts users to rate their driver and leave feedback about their ride experience. User ratings and feedback contribute to the overall quality of service and help users make informed decisions when choosing a driver.

9. **Left a Review:** In addition to rating the driver, users have the option to provide a written review of their ride experience. These reviews offer valuable insights into user satisfaction and areas for improvement.

Typical Drop-Off Points in the Funnel

In any customer funnel, there are points at which users may drop off or abandon the process. Identifying these drop-off points is crucial for optimizing the funnel and improving the overall user experience. Metrocar's funnel is no exception, and we will explore potential drop-off points in later chapters of this report.

Robert Tiger

Understanding Metrocar's customer funnel and recognizing where users may encounter issues or challenges is the first step in our funnel analysis. In the subsequent chapters, we will dive deeper into data analysis to uncover insights and recommendations for addressing these challenges.

TL;DR / Summary:

Funnel Conversion Rates Analysis:

1. **App Download to Signup Conversion:**
   - Conversion Rate: 74%
   - Challenge: 26% of app downloaders don't complete signup.
   - **Recommendation:** Enhance registration and onboarding for a smoother user experience.

2. **Signup to Request Ride Conversion:**
   - Conversion Rate: 70%
   - Challenge: 30% of signups don't progress to ride requests.
   - **Recommendation:** Collect user feedback, improve usability, and offer incentives for first-time riders.

3. **Driver Acceptance to Payment Conversion:**
   - Conversion Rate: 51%
   - Challenge: Approximately 50% of users don't complete payment after their ride requests are accepted.
   - **Recommendation:** Review pricing and payment processes, enhance transparency, and address trust and security concerns.

4. **Ride to Review Conversion:**
   - Conversion Rate: 70%
   - Challenge: 30% of users who complete a ride don't leave a review.
   - **Recommendation:** Implement incentives for reviews, streamline the review process, and collect user feedback on their review experience.

Platform Level User Base

iOS:
- Strong user adoption with 14,290 App Downloads.
- Impressive conversion rates across stages, indicating high user engagement.
- **Challenge:** Notable drop-off from "Rides Accepted" to "Pickups," indicating room for improvement.

Android:
- Moderate user adoption with 6,935 App Downloads.
- Strong conversion rates, similar to iOS.
- **Challenge:** Significant drop-off at the "Pickups" stage, requiring investigation and improvements.

Web:
- Smallest user base with 2,383 App Downloads.
- Lower conversion rates at each stage compared to iOS and Android.
- **Challenge:** Drop-offs at various stages suggest the need for user experience improvements.

Recommendations for Marketing Budget Focus:

1. **iOS:** Focus on retaining and engaging iOS users. Run loyalty and referral programs.
2. **Android:** Invest in user acquisition and address the "Pickups" drop-off.
3. **Web:** Prioritize usability and user experience improvements before scaling marketing efforts.

Age Range Analysis
- Uncovering an unnoticed group of users labeled "Null" who downloaded the app without specifying age.
- **Recommendation:** Thoroughly examine the signup process to understand why users download the app but don't complete registration.
- Analyze various age groups and their contributions to the funnel.
- Notable drop from "Rides Accepted" to "Pickups," especially in the 18-24 age group.
- **Recommendation:** Enhance the transition from "Rides Accepted" to "Pickups."

Analyzing Ride Request Distribution for Surge Pricing Strategy
- Understand the distribution of ride requests throughout the day for surge pricing strategy.
- Identify peak hours and off-peak hours.
- **Recommendations:** Implement data-driven pricing, adjust pricing during peak hours, and provide transparent user communication.

Improving Conversion Rates in the Funnel
- Identify the stage with the lowest conversion rate: "Driver Acceptance" to "Pickups."
- Investigate potential factors causing low conversion.
- **Recommendation:** Implement real-time driver communication, improve user interface, and gather user feedback for improvements.

Chapter 1: Identifying Funnel Drop-off Points and Improvement Recommendations

In this chapter, we address the critical business question of pinpointing drop-off points in Metrocar's customer funnel and recommending areas for improvement. Our analysis draws from the provided data, focusing on the key funnel stages that play a significant role in users' journey towards completing their ride.
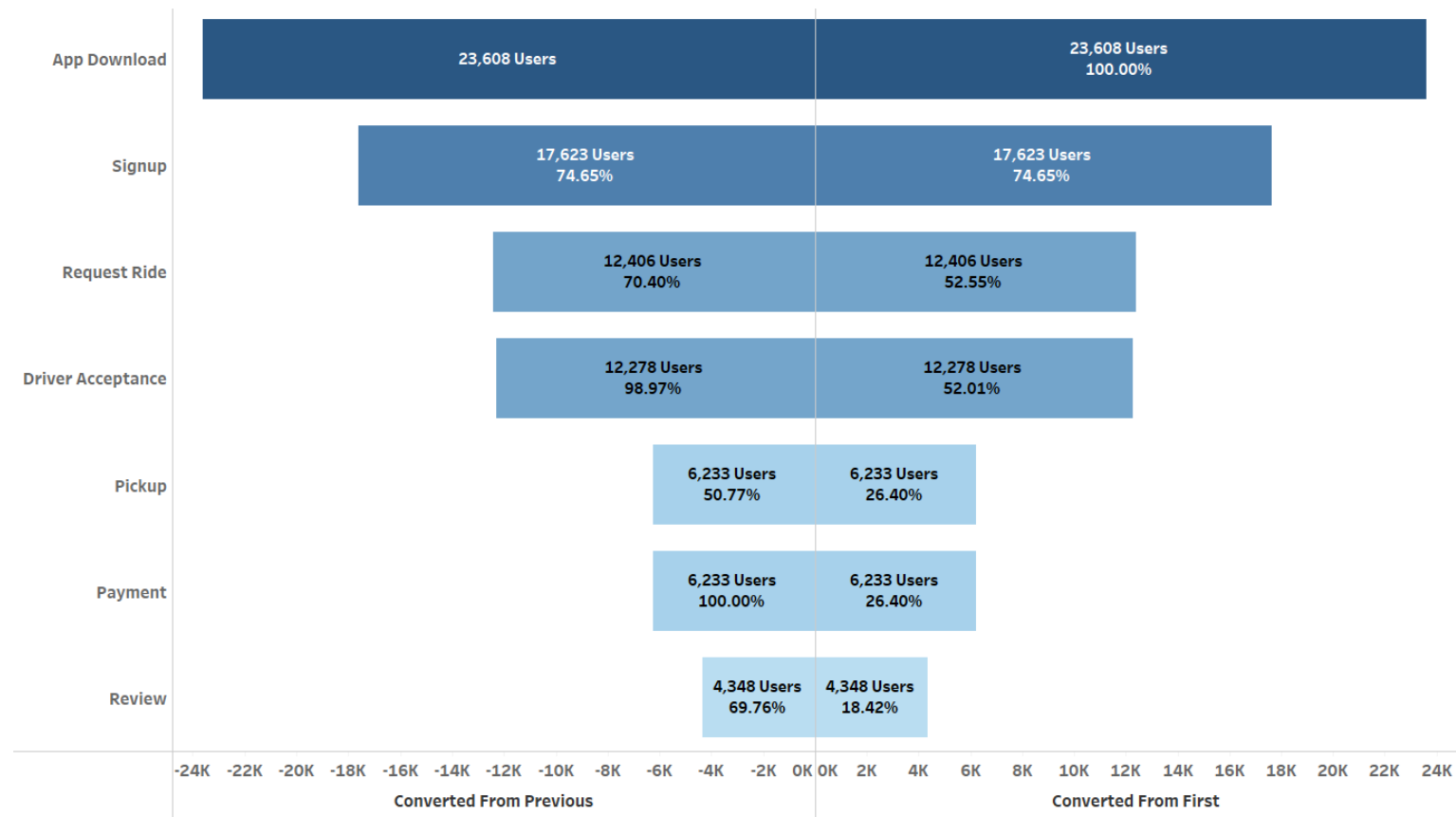
Funnel Conversion Rates Analysis

1. App Download to Signup Conversion:

The conversion rate from App Download to Signup is approximately **74%**. While this is a relatively high conversion rate, it suggests that approximately **26%** of users who downloaded the app did not complete the signup process. Potential reasons for this drop-off include:

- Barriers during the registration or onboarding process.
- User confusion or difficulties during the signup phase.
- Lack of motivation or value proposition for users to complete registration.

Recommendations: We should conduct user testing to identify pain points during registration and onboarding. Make the process more seamless, informative, and engaging.

2. Signup to Request Ride Conversion:

The conversion rate from Signup to Request Ride is approximately **70%**. Around **30%** of users who signed up did not proceed to request a ride. Possible issues that may contribute to this drop-off include:

- Usability concerns in our app's interface or workflow.

- User concerns regarding the safety or reliability of our services.
- Inadequate motivation or incentives for users to book their first ride.

Recommendations: User feedback should be collected to understand the reasons behind this drop-off. Address usability issues, enhance user trust, and consider providing incentives for first-time users.

3. Driver Acceptance to Pickup Conversion:

The conversion rate from <u>Driver Acceptance</u> → <u>Pickup</u> is approximately **51%**. This indicates that approximately half of the users who have had their ride requests accepted do not proceed to getting picked up. Potential factors contributing to this drop-off include:

- Pricing concerns or a lack of transparency in pricing after accepting a ride.
- Long wait time for drivers.
- Uncertainty with the communications with the driver en route.

Recommendations: Conduct a thorough review of pricing and payment processes. Ensure transparency and flexibility in payment options. Address any trust and security issues related to payment.

4. Ride to Review Conversion:

The conversion rate from Ride to Review is approximately **70%**. While this conversion rate is relatively high, it's important to note that roughly **30%** of users who completed a ride did not leave a review. The drop-off might be due to:

- Lack of motivation or incentives for users to provide feedback.
- Complexity or friction in the review submission process.

Recommendations: Consider implementing incentives for users to leave reviews. Streamline the review submission process and gather user feedback on their review experience.

Chapter 2: Platform Level

User Base
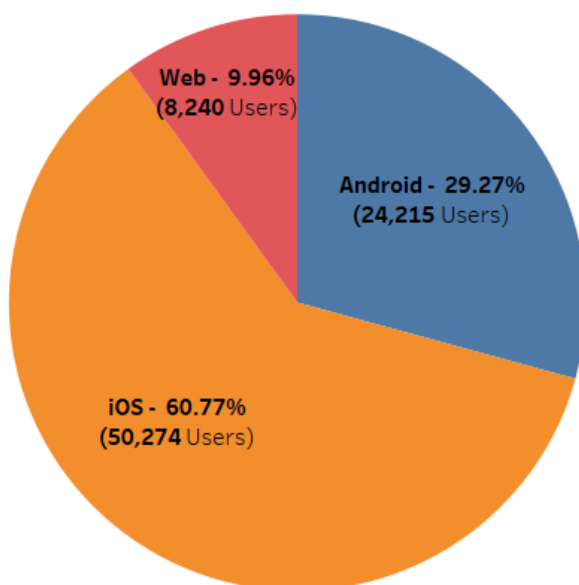
We have a substantial user base across all platforms.

- **iOS** stands out as the platform with the largest user count, with **50,724 users (60% of total users)**.
- **Android** is a strong second with **24,215 users (~30% of total users)**
- **Web** has **8,240 users (10% of total users).**

Robert Tiger

Key Findings by Platform

iOS

1. **High User Adoption:** iOS leads with **14,290 App Downloads**, significantly more than Android and Web.

2. **Conversion Rates:** Conversion rates on iOS are impressive. The App Downloads → Signups conversion rate is about **75%**, Signups → Ride Requests is **70%**, and Ride Requests → Rides Accepted is **99%.**

3. **Drop-off at Pickups:** There's a **significant drop** of **48%** from Rides Accepted → Pickups, indicating a need to address this point in the funnel.

4. **Reviews Engagement:** The Payment → Reviews conversion rate is **69%**, showing a strong tendency among iOS users to leave reviews.

	Converted from Previous	Converted from First
App Downloads	14,290 Users	14,290 Users 100.00%
Signups	10,728 Users 75.07%	10,728 Users 75.07%
Ride Requests	7,550 Users 70.38%	7,550 Users 52.83%
Rides Accepted	7,471 Users 98.95%	7,471 Users 52.28%
Pickups	3,792 Users 50.76%	3,792 Users 26.54%
Payment	3,792 Users 100.00%	3,792 Users 26.54%
Reviews	2,651 Users 69.91%	2,651 Users 18.55%

Robert Tiger
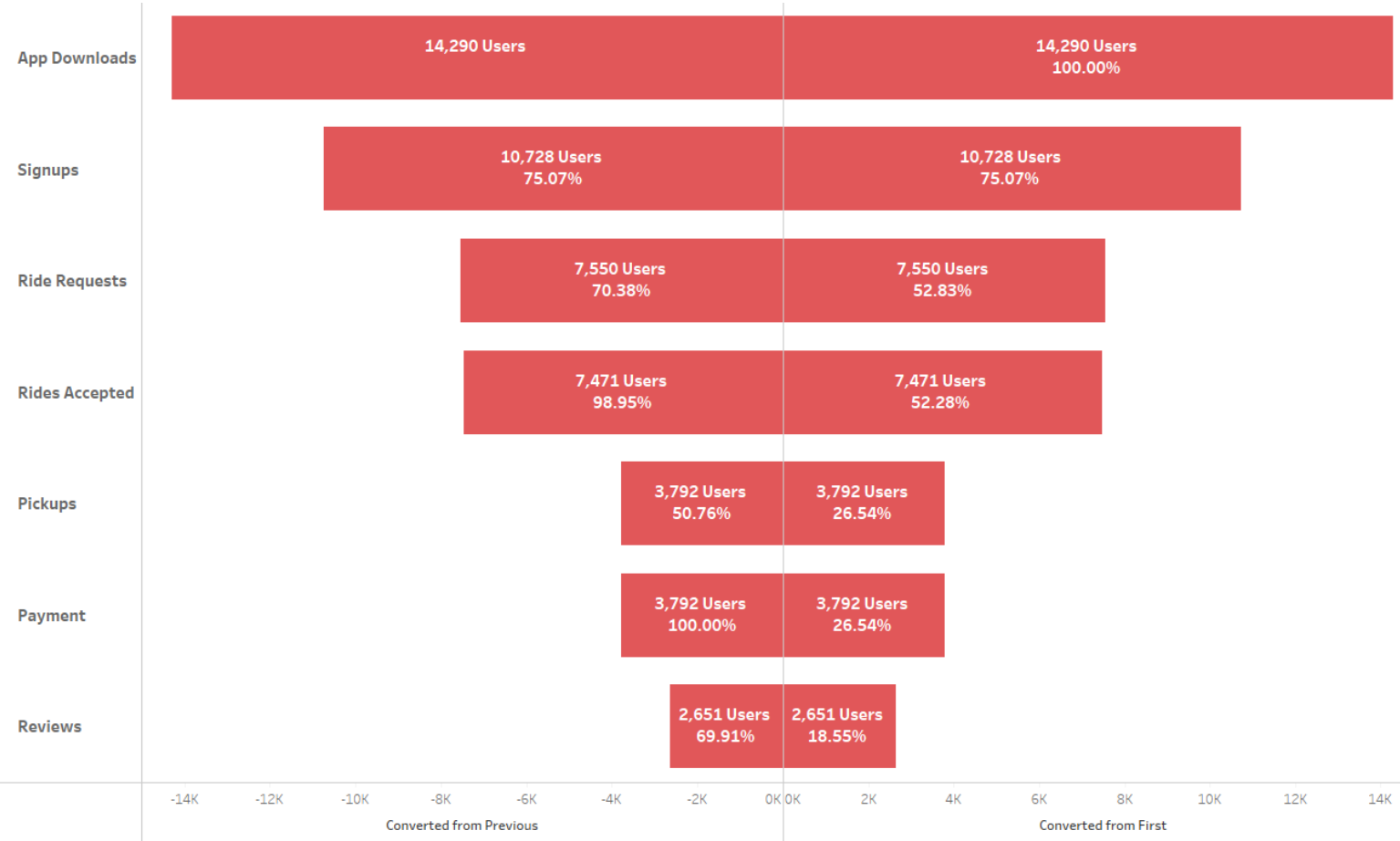
Android

1. **Moderate User Adoption**: Android has **6,935 App Downloads**, less than iOS but still substantial.

2. **Conversion Rates**: Android also exhibits robust conversion rates. The App Downloads → Signups conversion rate is about **74%**, Signups → Ride Requests is **70%**, and Ride Requests → Rides Accepted is **99%**.

3. **Drop-off at Pickups**: Similar to iOS, there's a significant **49%** drop at Pickups. Investigating and addressing this issue is crucial.

4. **Reviews Engagement**: Android users have a **69%** conversion rate from Payment → Reviews, indicating a willingness to provide feedback.

	Converted from Previous	Converted from First
App Downloads	6,935 Users	6,935 Users 100.00%
Signups	5,148 Users 74.23%	5,148 Users 74.23%
Ride Requests	3,619 Users 70.30%	3,619 Users 52.18%
Rides Accepted	3,580 Users 98.92%	3,580 Users 51.62%
Pickups	1,830 Users 51.12%	1,830 Users 26.39%
Payment	1,830 Users 100.00%	1,830 Users 26.39%
Reviews	1,273 Users 69.56%	1,273 Users 18.36%

Robert Tiger
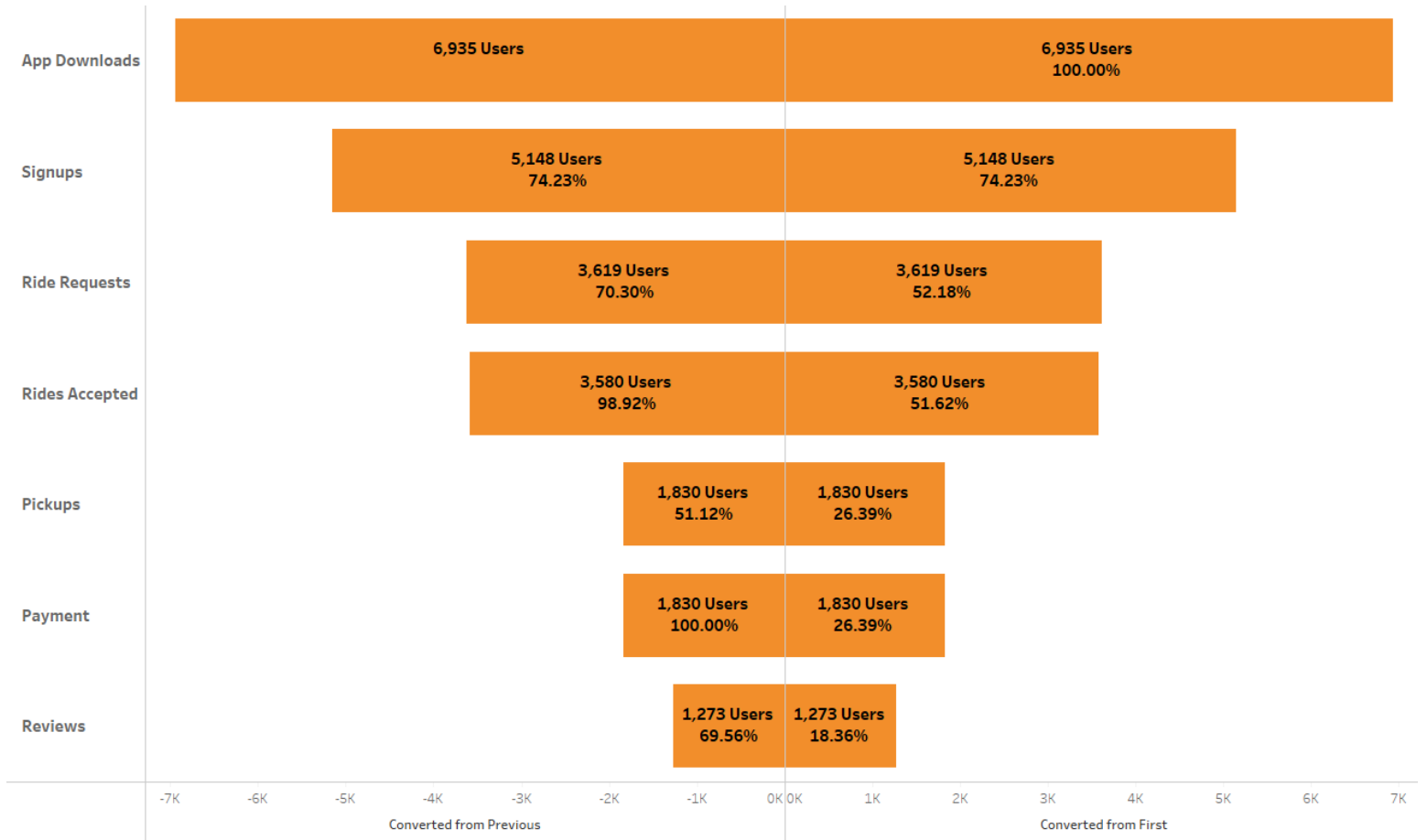
Web

1. **Smallest User Base**: The Web platform has the fewest App Downloads at **2,383**, suggesting room for growth.

2. **Conversion Rates**: Web has lower conversion rates at each stage of the funnel compared to iOS and Android. App Downloads → Signups is around **73%**, Signups → Ride Requests is **71%**, and Ride Requests → Rides Accepted is **99%**. This indicates potential usability or user experience issues.

3. **Drop-off at Various Stages**: There are drop-offs at various stages of the funnel, potentially indicating the need for user experience improvements.
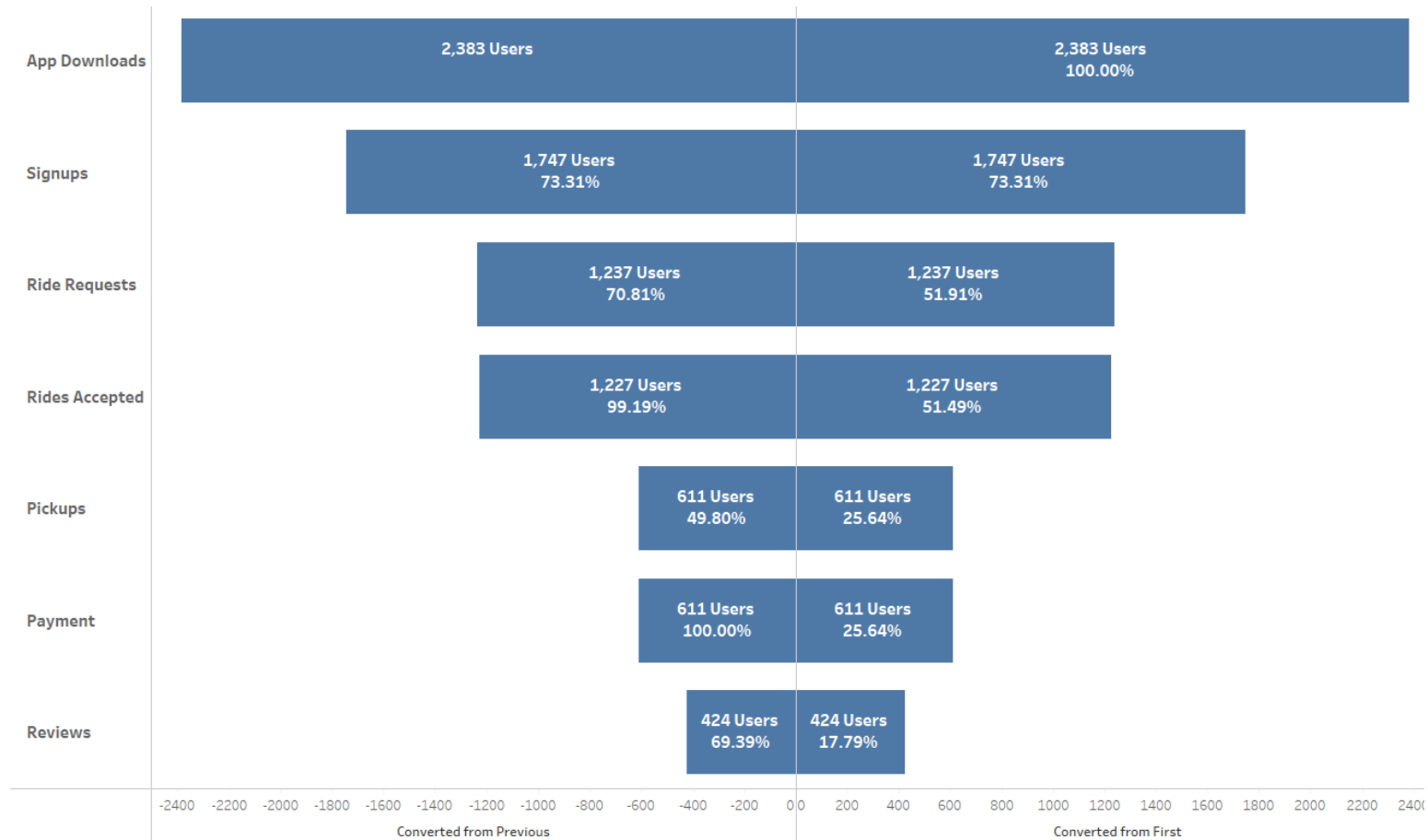

Recommendations for Marketing Budget Focus

1. **iOS**: With a strong user base and healthy conversion rates, focusing marketing efforts on retaining and engaging iOS users is recommended. Consider running loyalty and referral programs to take advantage of the **69%** Payment → Reviews conversion rate.

2. **Android**: Android has a solid user base and engagement. It's essential to invest in user acquisition and address the **48%** drop-off at Pickups to enhance the user experience.

3. **Web**: While Web has the smallest user base and lower conversion rates, focus on identifying and resolving usability and user experience issues before scaling marketing efforts. The platform's high Ride Requests → Rides Accepted conversion rate (**99%**) suggests potential interest, but user experience improvements are needed.

Balancing marketing efforts based on these findings and continuous monitoring and optimization of funnel metrics are key to achieving long-term success across all platforms.

Robert Tiger

Chapter 3: Age Range Analysis
User Base
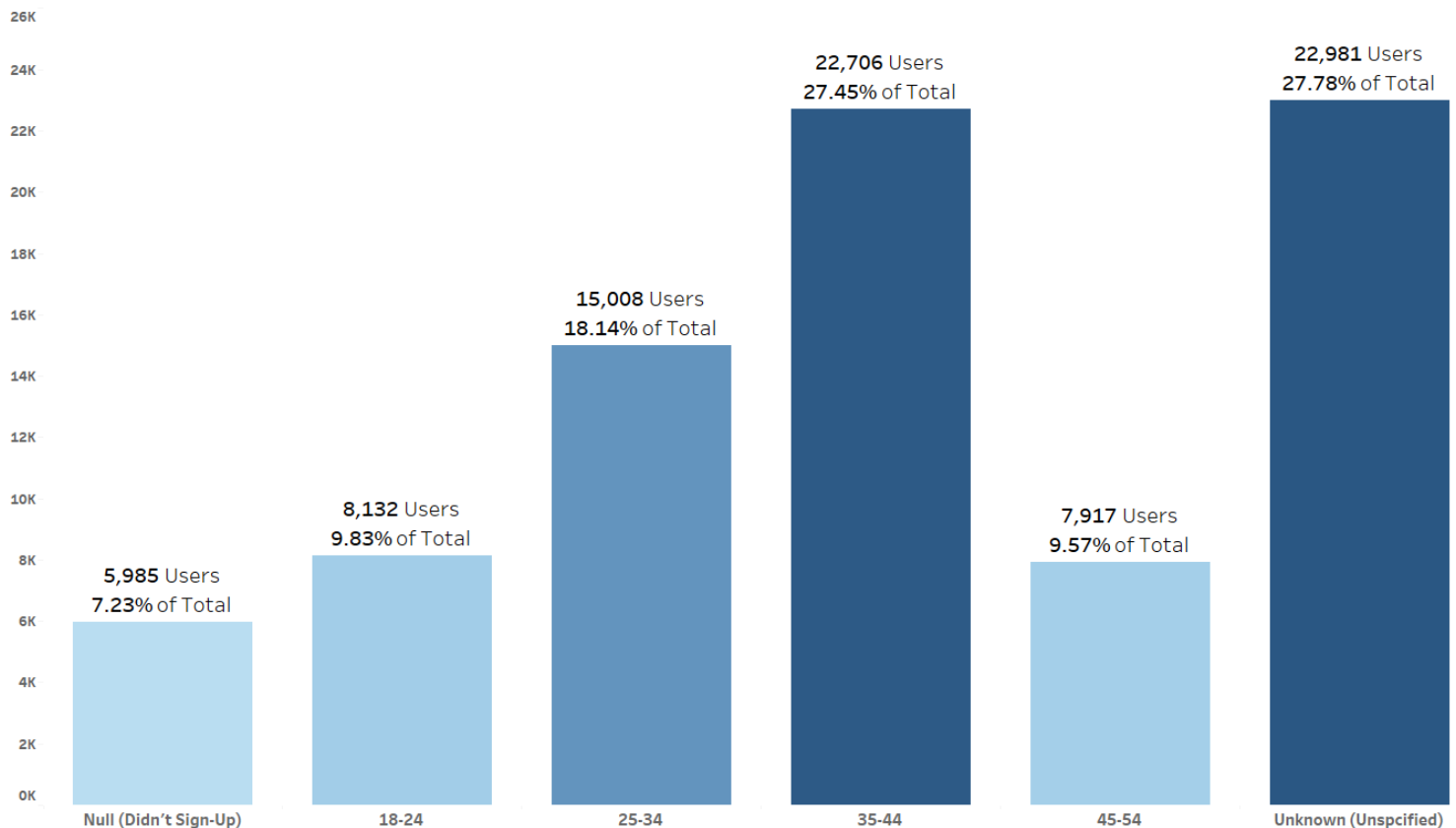
Reevaluating Funnel Analysis: Addressing Null Age Groups

Our funnel analysis has revealed a significant group of users, totalling **5,985**, who downloaded the app without specifying their age—labelled as *"Null"* age group. This has led to a seemingly perfect 100% conversion rate from App Downloads → Signups. **However**, closer inspection reveals that these users did not proceed to sign up, and therefore don't have age registered, highlighting an uncharted territory within the initial stages of user engagement. To address this, our recommendation is to thoroughly examine the signup process to understand why users download the app but do not complete the registration. This examination is essential for a more accurate representation of our funnel and to improve user acquisition strategies. Luckily enough, those Null users are at the bottom of our users' pool (accounting for only **7.23%** of the total users) and the majority of users proceeded to complete the registration and enter age.

For users who DID signup and enter an age:

- **18-24**: The youngest age range encompasses **8,132** users, reflecting a portion of our tech-savvy, youthful user base.

- **25-34**: This age group comprises **15,008** users, positioning them as a substantial component of our user community.

- **35-44**: The age range of 35-44 represents the **largest group**, with **22,706** users. This demographic stands out as a critical segment of our user base and will be examined in detail throughout this chapter.

- **45-54**: **7,917** users fall within this age range, illustrating a robust presence of users in the mid-range demographic.

- **Unknown (Unspecified)**: The Unknown age group, with **22,981** users, constitutes a significant portion of our user base. Understanding and engaging this demographic is of paramount importance.

Key Findings: Age Group Analysis in the Funnel

1. **App Downloads**: The **35-44** age group leads with **5,181** app downloads, followed by the **25-34** age group with **3,447**. **Unknown** age and **18-24** groups also make significant contributions.

2. **Ride Requests**: The **35-44** age group leads with **3,662** ride requests, followed by the **25-34** age group with **2,425**. There's a <u>significant drop</u> from <u>app downloads</u> → <u>ride requests</u> in the **18-24** age group, which could be a focus for improvement.

3. **Rides Accepted**: The **35-44** age group remains dominant with **3,628** rides accepted, followed by the **25-34** age group with **2,393**. This stage shows a <u>high conversion rate</u>.

4. **Pickups, Payment, and Reviews**: The **35-44** age group continues to lead, with **1,861** pickups, **1,861** payments, and **1,332** reviews. The **25-34** age group closely follows.
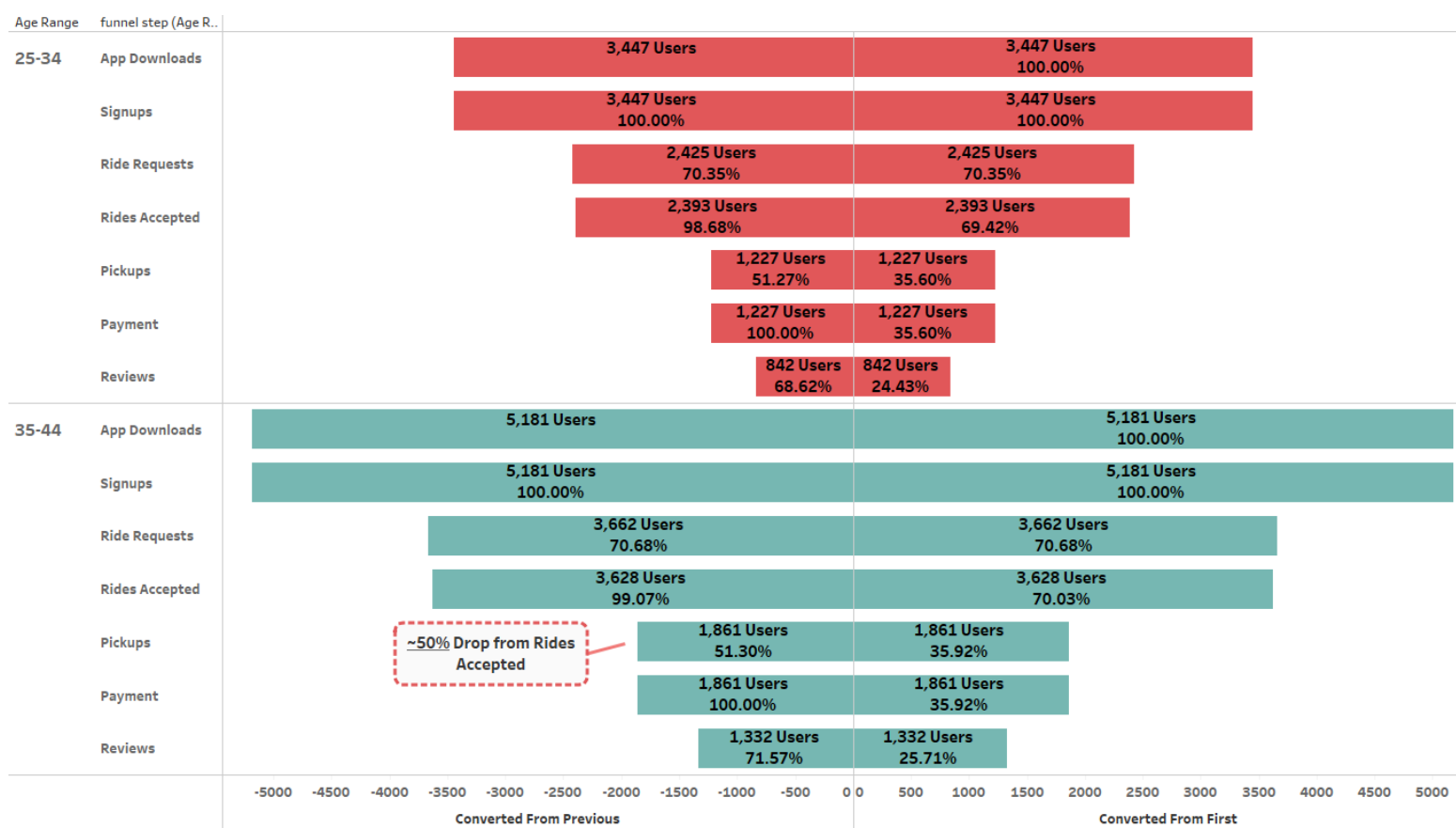

Major Conversion Rates and Drop-offs

- **Signups → Ride Requests (Drop-off)**: <u>A significant drop</u> occurs at this stage, especially in the **18-24** age group, which experiences a drop of **~30%.**

- **Ride Requests → Rides Accepted**: The conversion rate remains high across all age groups, at the high **90's%**, showing user commitment to ride acceptance.

- **Pickups → Payment:** All rides where customers were picked up by the driver, finished without any payment issues. **100%** of the users that took a ride completed the payment process.

- **Leaving a Review:** There is about **30% drop**, in all age groups, from when a user paid for the ride to actually leaving a review.

The Drop from Rides Accepted → Pickups Across Age Groups

- In the "Rides Accepted" stage, users across all age groups show a strong commitment to accept rides. For example, in the **35-44** age group, we have **3,628** users accepting rides.

- However, when we progress to the "Pickups" stage, we notice a significant drop in user engagement. In the same **35-44** age group, we have **3,662** users at the "Ride Accepted" stage and **1,861** users at the "Pickups" stage, indicating a conversion rate of only half, **51%.** This conversion rate is indeed sub-optimal.

Robert Tiger
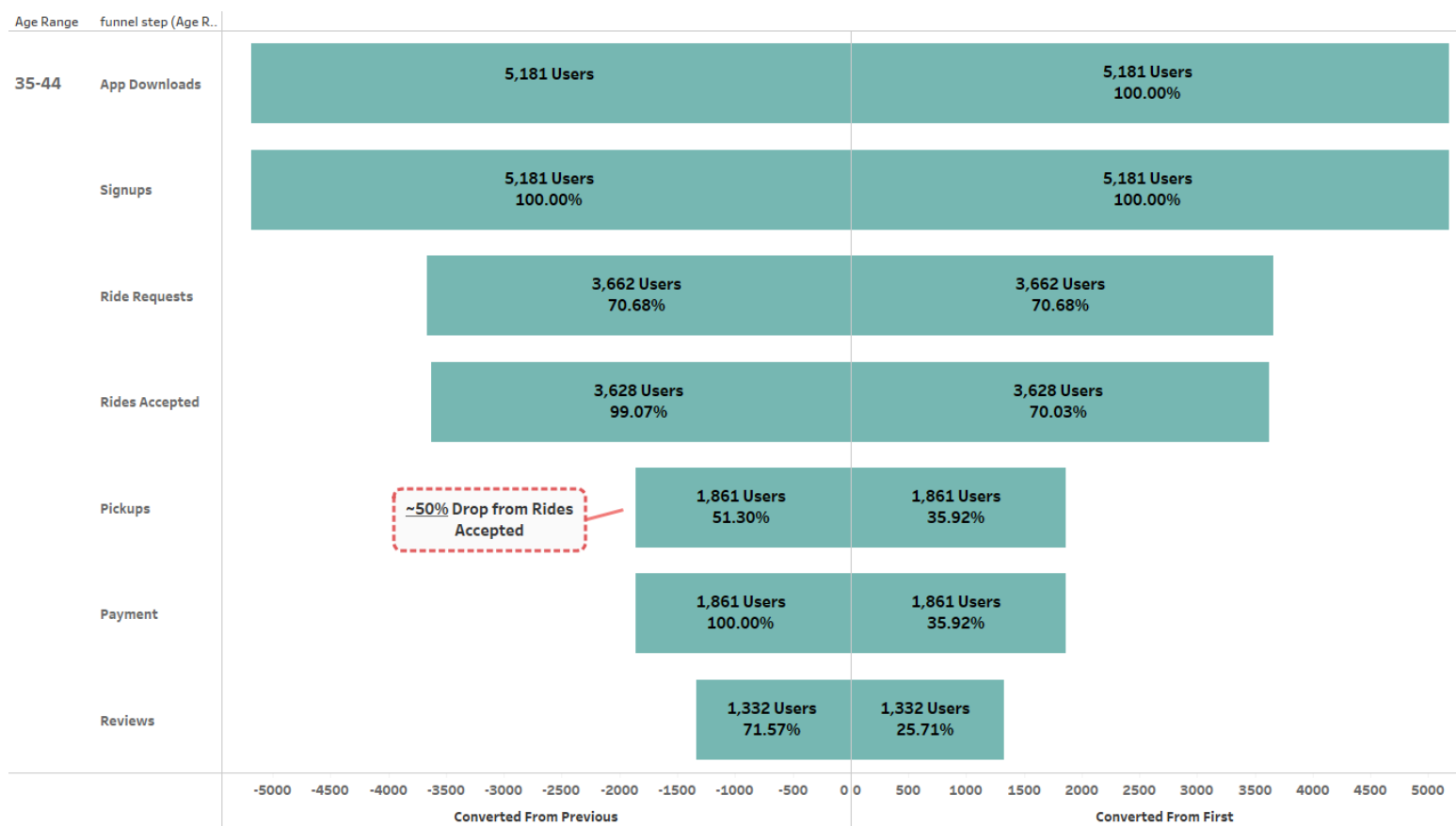
Let's examine another age group as well:

- In the **18-24** age group, **1,289** users accept rides, but only **670** users proceed to pickups, resulting in a conversion rate of **52%.**

Robert Tiger
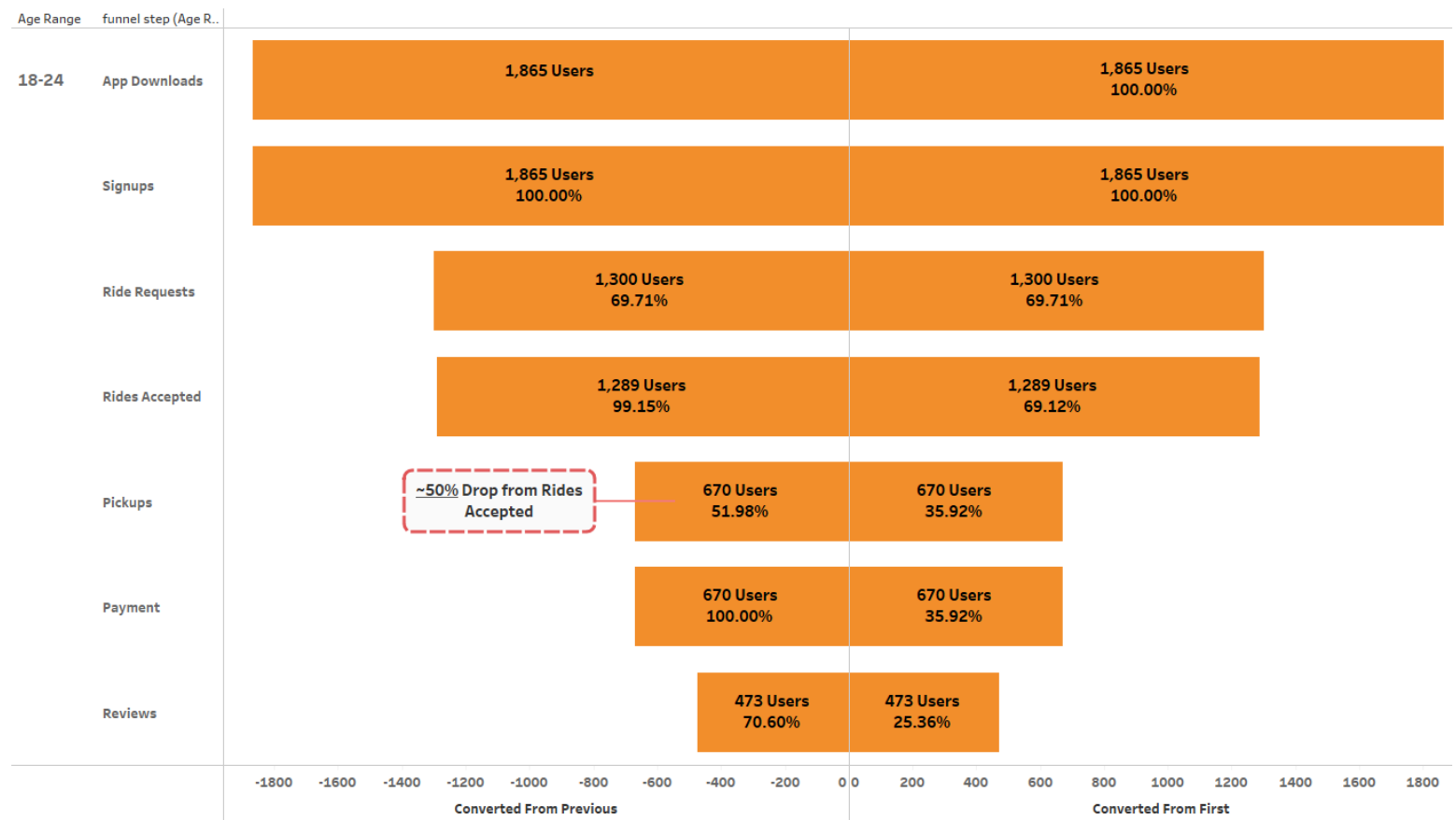
Recommendation: Enhancing the Transition from Rides Accepted → Pickups

To improve the conversion rate from "Rides Accepted" → "Pickups," we recommend the following measures:

1. Real-time Driver Communication:
- Implement a real-time communication system between drivers and users. This enables users to track their driver's location and estimated arrival time, reducing uncertainty and anxiety associated with waiting.

2. Timely Notifications:
- Send proactive notifications to users, providing updates on their driver's progress and expected arrival time. This information empowers users to plan their pickup more efficiently and minimizes idle waiting.

3. Driver Scheduling and Predictability:
- Enhance driver scheduling algorithms to ensure more predictable pickup times. Users are more likely to complete the journey if they can rely on consistent and timely pickups.

4. In-App Features:
- Incorporate in-app features that allow users to specify their preferred pickup time, helping to align user expectations with driver availability.

5. Driver Incentives:
- Implement incentives for drivers to minimize wait times and ensure efficient pickups, creating a win-win situation for both drivers and users.

Identifying Target Customer Age Groups

25-34 Age Group:
- This age group consists of **15,008 users**, making it a substantial portion of Metrocar's user base. The strong conversion rates observed across the funnel stages, coupled with the significant user count, suggest that users between the ages of 25-34 are not only actively engaging with the platform but also form a sizable customer base.

35-44 Age Group:
- With **22,706** users in the 35-44 age group, it represents the largest user segment. While the conversion rates in this age group are consistently high, the absolute number of users in this category indicates a substantial and potentially significant portion of Metrocar's customer base.

Considering both the conversion rates and the total user count, the **25-34** and **35-44** age groups are both highly relevant to Metrocar's target customer base. These two age groups exhibit strong user engagement and are numerically significant, making them prime candidates for tailored marketing strategies and service enhancements. A more detailed user profiling and market research approach can help fine-tune these insights and refine the targeting efforts for the most effective results.

Robert Tiger

Chapter 4: Analysing Ride Request Distribution for Surge Pricing Strategy

Introduction: Surge Pricing in Ride-Sharing

Surge pricing, the practice of adjusting fares in response to demand, plays a crucial role in the ride-sharing industry. This chapter explores the distribution of ride requests throughout the day to inform the development of an effective surge pricing strategy. By understanding how ride requests vary across different hours, we can tailor our pricing strategy to match user demand patterns.

Breakdown of Ride Request Patterns by Hour

Peak Hours:

1. **Morning Rush Hours (8 AM and 9 AM):** The day begins with a surge in demand, with **60,071** and **60,210** ride requests, respectively, at 8 AM and 9 AM. This period corresponds to the start of the workday, indicating that users heavily rely on the service for their daily commutes.
2. **Afternoon Rush Hours (4 PM and 5 PM):** Another significant demand peak occurs in the late afternoon, with **58,527** and **58,176** ride requests at 4 PM and 5 PM, respectively. This aligns with the end of the workday and suggests that users are seeking rides as they finish their professional commitments.
3. **Evening Hours (6 PM to 8 PM):** Demand remains substantial in the early evening, with counts ranging from **40,372** to **39,495**. Users are actively using the service for various purposes, including commuting back home or for evening outings.

Off-Peak Hours:

1. **Late Evening and Early Morning (1 AM to 5 AM):** The late evening and early morning hours exhibit lower demand, with the counts reaching their lowest points during this period. Users are less likely to request rides during these late-night and early morning hours.

2. **Mid-Day and Early Afternoon:** While there is consistent demand throughout the day, the hours between **10 AM** and **3 PM** show moderate demand, with counts ranging from **7,928** to **9,072**. These are the off-peak hours when users might not be engaged in daily commutes.

Num of Requests and Payments by Hour
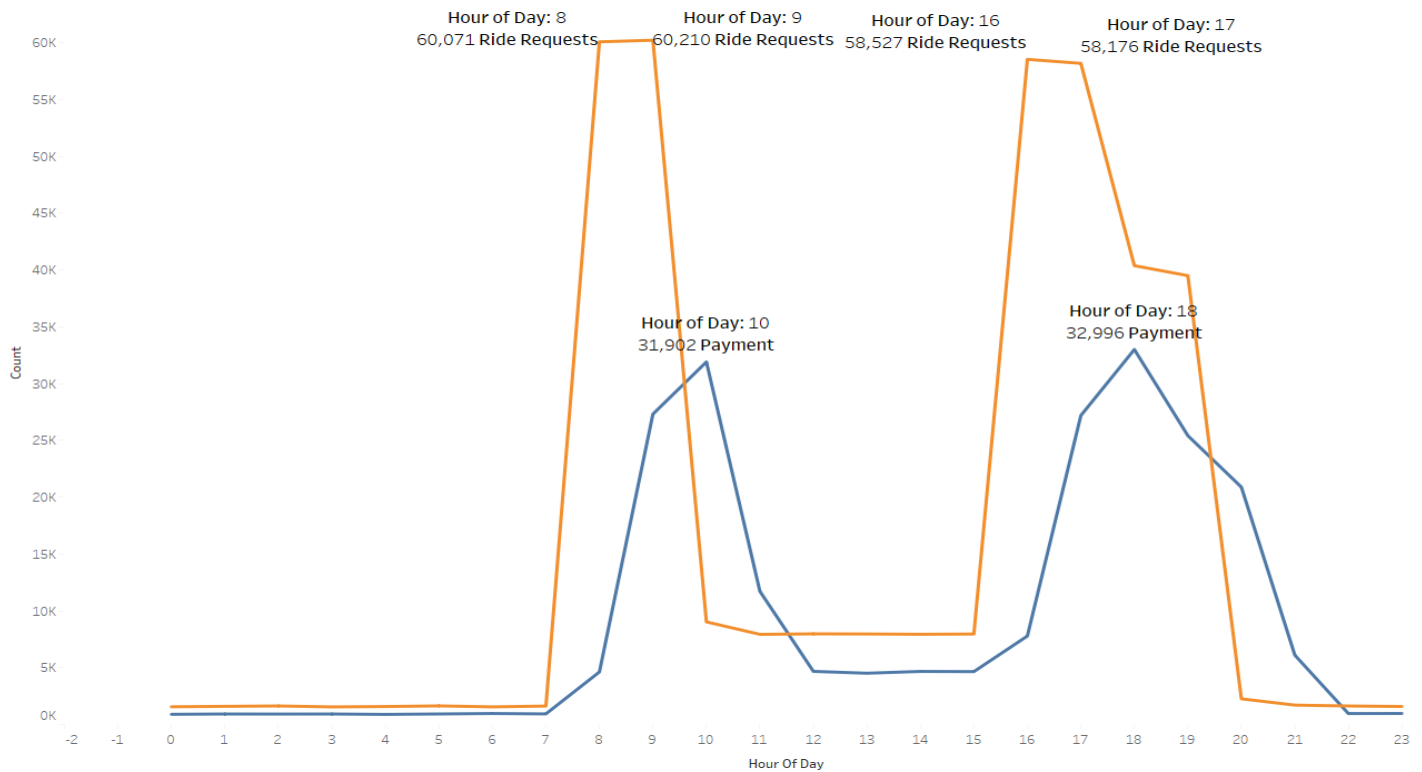
Recommendations for Surge Pricing Implementation

1. **Data-Driven Pricing Algorithms:** It is recommended that we implement data-driven pricing algorithms that take into account the hourly patterns and demand peaks identified in the previous analysis. These algorithms should adjust pricing in real-time to match supply and demand effectively.
2. **Peak Hour Special Rates:** We should consider assigning surge pricing rate multipliers during peak hours (e.g., morning and afternoon rush hours) to reflect increased demand. For instance, during these peak periods, fares can be increased to balance supply and demand.
3. **Gradual Pricing Adjustments:** We should aim to avoid sudden and drastic price hikes, as this can lead to user dissatisfaction. Instead, we should consider implementing gradual adjustments to pricing, keeping it responsive to hourly demand.
4. **Transparent User Communication:** It is recommended that we clearly communicate surge pricing to users, ensuring transparency and clarity. We should notify users when surge pricing is in effect and provide fare estimates during these periods.
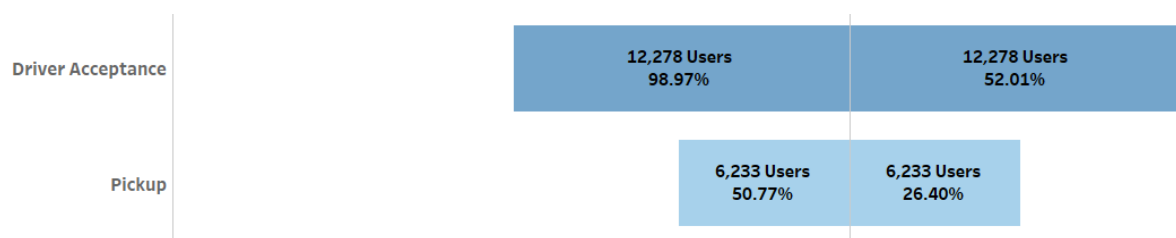
Chapter 5: Improving Conversion Rates in the Funnel

Introduction: Addressing Low Conversion Rates

This chapter delves into the analysis of the platform's funnel to identify the stage with the lowest conversion rate and explores strategies to enhance conversion at that stage.

Identifying the Funnel Stage with the Lowest Conversion Rate

In analyzing the conversion rates at each stage, we can clearly identify the stage with the lowest conversion rate. Comparing the number of users who successfully transitioned from one stage to the next, we find that "Pickups" stage has the lowest conversion rate, with **6,233** users out of the **12,406** who requested rides and **12,278** who accepted rides proceeding to the pickup stage.

Driver Acceptance	12,278 Users 98.97%	12,278 Users 52.01%
Pickup	6,233 Users 50.77%	6,233 Users 26.40%

The transition from the "Driver Acceptance" → "Pickups" stage experiences a considerable drop-off of approximately **49%**. Approximately **75%** drop-off from "Downloads" → "Pickups" (Only **26%** of users who downloaded the app reached the pickup stage)

Investigating Potential Factors Contributing to Low Conversion

To address the low conversion rates at the identified stages of the user journey, it is essential to explore the factors that contribute to these challenges.

Factors to Investigate:

1. **User Feedback:** We should gather and analyze user feedback, reviews, and complaints related to the "Pickups" stage. Understanding the specific pain points and frustrations expressed by users can offer valuable insights.

2. **Usability Issues:** We need to examine the user interface and experience at these stages to identify any usability issues, complications, or barriers that may discourage users from proceeding.

3. **Communication and Transparency:** It's essential to assess the clarity and transparency of communication at these stages, particularly with regard to pricing, payment methods, and waiting times.

4. **Competitor Benchmarking:** We should consider how competitors handle these stages and whether there are best practices or features that we can adopt to enhance the user experience.

Recommendations for Enhancing Lowest Conversion Rate Stage

To address the significant drop-off observed in the transition from the "Driver Acceptance" → "Pickups" stage, it is essential to devise effective strategies that can improve the conversion rate at this critical juncture.

Data-Driven Recommendations:

1. **Real-Time Driver Location:** Implement real-time tracking and display of driver location to provide users with accurate information on the driver's arrival. This can reduce uncertainty and anxiety during the pickup process.

2. **Improved Communication:** Enhance communication channels between users and drivers. Consider options for users to contact drivers directly for any specific requirements or to coordinate pickup details effectively.

3. **In-App Alerts:** Utilize in-app notifications to inform users when their ride is accepted and to provide estimated pickup times. This ensures that users are well-informed and prepared for their ride.

4. **User Feedback Integration:** Encourage users to provide feedback on driver performance during the pickup stage, allowing for continuous improvement in service quality.

5. **Streamlined Payment Handling:** Ensure that payment processing and fare estimates are handled seamlessly. Users should have a clear understanding of the expected cost of the ride before and after the pickup.

6. **A/B Testing:** Implement A/B testing to evaluate the impact of changes made to the user interface, communication features, or other aspects aimed at addressing user challenges in this transition.

Enhancing the User Experience:

1. **Simplified User Interface:** Optimize the user interface to make it more intuitive and user-friendly, ensuring a seamless transition from driver acceptance to pickups.

2. **User Education:** Provide users with clear instructions and tips for the pickup process, guiding them on what to expect and how to coordinate effectively with the driver.

3. **Driver Training:** Invest in driver training programs to ensure drivers are equipped with the necessary skills and knowledge to offer a smooth pickup experience.

Ongoing Monitoring and Feedback:

1. **Data Analysis:** Continuously monitor data related to the transition from driver acceptance to pickups to assess the impact of implemented changes and identify areas for further improvement.

2. **User Feedback:** Encourage users to share their experiences and suggestions to refine the pickup process continually.

Robert Tiger

Appendix:

User Level Interactive Funnel Dashboard:

https://public.tableau.com/views/InteractiveDasboard_16975368891870/Dashboard1?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link

Workbook and Dashboard with 3 Funnels (User Level, Platform, Age Range):

https://public.tableau.com/views/Funnel_16963607142170/Dashboard1?:language=en-US&publish=yes&:display_count=n&:origin=viz_share_link

YouTube Link for Video Presentation:

https://youtu.be/spSxo32PyVs

SQL Queries Used:

-- How many times was the app downloaded?
SELECT COUNT(download_ts) AS downloads
FROM app_downloads

-- How many users signed up on the app?
SELECT COUNT(user_id) AS sign_ups
FROM signups

-- How many rides were requested through the app?
SELECT COUNT(*) AS ride_requests
FROM ride_requests
WHERE request_ts IS NOT NULL

-- How many rides were requested and completed through the app?
SELECT
  (SELECT COUNT(*) FROM ride_requests) AS ride_requests,
  (SELECT COUNT(*) FROM ride_requests
   WHERE pickup_ts IS NOT NULL AND dropoff_ts IS NOT NULL) AS rides_completed;

-- How many rides were requested and how many unique users requested a ride?
SELECT
  COUNT(*) AS total_rides_requested,
  COUNT(DISTINCT user_id) AS unique_users_requesting_ride
FROM ride_requests;

-- What is the average time of a ride from pick up to drop off?
SELECT
  ROUND(AVG(EXTRACT(EPOCH FROM (dropoff_ts - pickup_ts)))::NUMERIC, 2) AS average_ride_time_seconds,
  ROUND(AVG(EXTRACT(EPOCH FROM (dropoff_ts - pickup_ts)) / 60)::NUMERIC, 2) AS average_ride_time_minutes
FROM ride_requests
WHERE dropoff_ts IS NOT NULL AND pickup_ts IS NOT NULL;

-- How many rides were accepted by a driver?

Robert Tiger


```sql
SELECT COUNT(accept_ts) AS rides_accepted
FROM ride_requests
WHERE accept_ts IS NOT NULL;
```

-- How many rides did we successfully collect payments and how much was collected?
```sql
SELECT (SELECT COUNT(*)
    FROM transactions
    WHERE charge_status = 'Approved') AS approved_payments,
    (SELECT ROUND(SUM(purchase_amount_usd)::NUMERIC, 2) AS total_collected_usd
     FROM transactions WHERE charge_status = 'Approved')
FROM transactions
LIMIT 1;
```

-- How many ride requests happened on each platform?
```sql
SELECT platform, COUNT(request_ts) AS ride_requests
FROM metrocar
GROUP BY platform
ORDER BY 2 DESC
```

-- What is the drop-off from users signing up to users requesting a ride?
```sql
SELECT
  COUNT(DISTINCT user_id) AS signups,
  (SELECT COUNT(DISTINCT user_id) FROM metrocar WHERE request_ts IS NOT NULL) AS requests,
  (COUNT(DISTINCT user_id) - (SELECT COUNT(DISTINCT user_id) FROM metrocar WHERE request_ts
IS NOT NULL)) AS dropoff_count,
  ((COUNT(DISTINCT user_id) - (SELECT COUNT(DISTINCT user_id) FROM metrocar WHERE request_ts
IS NOT NULL)) / COUNT(DISTINCT user_id)::numeric) AS dropoff_rate
FROM metrocar
WHERE signup_ts IS NOT NULL
LIMIT 1;
```

-- User Level Funnel By Steps
```sql
WITH FunnelCTE AS (
  -- App Download
  SELECT
    ad.app_download_key::TEXT AS user_id,
    ad.platform,
    ad.download_ts AS funnel_ts,
    'App Download' AS funnel_step
  FROM
    app_downloads ad
  WHERE
    ad.platform IN ('ios', 'android', 'web')

  UNION ALL

  -- Signup
  SELECT
```

```
        s.user_id::TEXT,
        s.session_id::TEXT AS app_download_key,
        s.signup_ts AS funnel_ts,
        'Signup' AS funnel_step
    FROM
        signups s

    UNION ALL

    -- Request Ride
    SELECT
        rr.user_id::TEXT,
        s.session_id::TEXT AS app_download_key,
        rr.request_ts AS funnel_ts,
        'Request Ride' AS funnel_step
    FROM
        ride_requests rr
    JOIN
        signups s ON rr.user_id = s.user_id

    UNION ALL

    -- Driver Acceptance
    SELECT
        rr.user_id::TEXT,
        s.session_id::TEXT AS app_download_key,
        rr.accept_ts AS funnel_ts,
        'Driver Acceptance' AS funnel_step
    FROM
        ride_requests rr
    JOIN
        signups s ON rr.user_id = s.user_id
    WHERE
        rr.accept_ts IS NOT NULL

    UNION ALL

    -- Ride
    SELECT
        rr.user_id::TEXT,
        s.session_id::TEXT AS app_download_key,
        rr.pickup_ts AS funnel_ts,
        'Ride' AS funnel_step
    FROM
        ride_requests rr
    JOIN
        signups s ON rr.user_id = s.user_id
    WHERE
```

```
        rr.pickup_ts IS NOT NULL

    UNION ALL

    -- Payment
    SELECT
        s.session_id::TEXT AS user_id,
        s.session_id::TEXT AS app_download_key,
        t.transaction_ts AS funnel_ts,
        'Payment' AS funnel_step
    FROM
        transactions t
    JOIN
        ride_requests rr ON t.ride_id = rr.ride_id
    JOIN
        signups s ON rr.user_id = s.user_id

    UNION ALL

    -- Review
    SELECT
        r.user_id::TEXT,
        rr.user_id::TEXT AS app_download_key,
        rr.dropoff_ts AS funnel_ts,
        'Review' AS funnel_step
    FROM
        reviews r
    JOIN
        ride_requests rr ON r.ride_id = rr.ride_id
)
SELECT
    funnel_step,
    COUNT(DISTINCT user_id) AS step_count
FROM
    FunnelCTE
GROUP BY
    funnel_step
ORDER BY
    step_count DESC;

-- Platform Granuality
WITH FunnelCTE AS (
  SELECT
        platform,
        'App Downloads' AS funnel_step,
  COUNT(DISTINCT app_download_key) AS funnel_count
  FROM metrocar
  GROUP BY 1
```

Robert Tiger


```sql
  ORDER BY funnel_count DESC
),
SignupCTE AS (
 SELECT
 platform,
        'Signups' AS funnel_step,
        COUNT(DISTINCT user_id) AS funnel_count
 from metrocar
 WHERE signup_ts IS NOT NULL
        GROUP BY 1
 ORDER BY funnel_count DESC
),
RideRequestCTE AS (
   SELECT
                        platform,
     'Ride Requests' AS Funnel_Step,
     COUNT(DISTINCT session_id) AS Funnel_Count
   FROM metrocar
   WHERE request_ts IS NOT NULL
                GROUP BY 1
 ORDER BY funnel_count DESC
),
RidesAcceptedCTE AS (
   SELECT
     platform,
                        'Rides Accepted' AS Funnel_Step,
     COUNT(DISTINCT user_id) AS Funnel_Count
   FROM metrocar
   WHERE accept_ts IS NOT NULL
        GROUP BY 1
 ORDER BY funnel_count DESC
),
PickupsCTE AS (
   SELECT
                        platform,
     'Pickups' AS Funnel_Step,
     COUNT(DISTINCT user_id) AS Funnel_Count
   FROM metrocar
   WHERE pickup_ts IS NOT NULL
        GROUP BY 1
 ORDER BY funnel_count DESC
),
PaymentCTE AS (
   SELECT
                        platform,
     'Payment' AS Funnel_Step,
     COUNT(DISTINCT user_id) AS Funnel_Count
   FROM metrocar
```

Robert Tiger

```sql
  WHERE transaction_ts IS NOT NULL
                GROUP BY 1
  ORDER BY funnel_count DESC
),
ReviewsCTE AS (
  SELECT
                        platform,
    'Reviews' AS Funnel_Step,
    COUNT(DISTINCT user_id) AS Funnel_Count
  FROM metrocar
  WHERE review IS NOT NULL
        GROUP BY 1
  ORDER BY funnel_count DESC
)
SELECT
                platform,
  Funnel_Step,
  Funnel_Count
FROM FunnelCTE
UNION ALL
SELECT * FROM SignupCTE
UNION ALL
SELECT * FROM RideRequestCTE
UNION ALL
SELECT * FROM RidesAcceptedCTE
UNION ALL
SELECT * FROM PickupsCTE
UNION ALL
SELECT * FROM PaymentCTE
UNION ALL
SELECT * FROM ReviewsCTE
ORDER BY platform, funnel_count DESC;

-- Age Range Granulaity
WITH FunnelCTE AS (
 SELECT
 1 AS step,
        age_range,
        'App Downloads' AS funnel_step,
 COUNT(DISTINCT app_download_key) AS funnel_count
 FROM metrocar
 GROUP BY 1, 2
 ORDER BY 2
),
SignupCTE AS (
 SELECT
 2 AS step,
 age_range,
```

Robert Tiger

```sql
        'Signups' AS funnel_step,
        COUNT(DISTINCT user_id) AS funnel_count
  from metrocar
  WHERE signup_ts IS NOT NULL
        GROUP BY 1, 2
  ORDER BY 2
),
RideRequestCTE AS (
  SELECT
                        3 AS step,
                        age_range,
    'Ride Requests' AS Funnel_Step,
    COUNT(DISTINCT session_id) AS Funnel_Count
  FROM metrocar
  WHERE request_ts IS NOT NULL
                GROUP BY 1, 2
  ORDER BY 2
),
RidesAcceptedCTE AS (
  SELECT
                        4 as step,
    age_range,
                        'Rides Accepted' AS Funnel_Step,
    COUNT(DISTINCT user_id) AS Funnel_Count
  FROM metrocar
  WHERE accept_ts IS NOT NULL
        GROUP BY 1, 2
  ORDER BY 2
),
PickupsCTE AS (
  SELECT
                        5 as step,
                        age_range,
    'Pickups' AS Funnel_Step,
    COUNT(DISTINCT user_id) AS Funnel_Count
  FROM metrocar
  WHERE pickup_ts IS NOT NULL
        GROUP BY 1, 2
  ORDER BY 2
),
PaymentCTE AS (
  SELECT
                        6 as step,
                        age_range,
    'Payment' AS Funnel_Step,
    COUNT(DISTINCT user_id) AS Funnel_Count
  FROM metrocar
  WHERE transaction_ts IS NOT NULL
```

```
                  GROUP BY 1, 2
  ORDER BY 2
),
ReviewsCTE AS (
    SELECT
                        7 as step,
                        age_range,
        'Reviews' AS Funnel_Step,
        COUNT(DISTINCT user_id) AS Funnel_Count
    FROM metrocar
  WHERE review IS NOT NULL
        GROUP BY 1, 2
  ORDER BY 2
)
SELECT
                step,
                age_range,
    Funnel_Step,
    Funnel_Count
FROM FunnelCTE
UNION ALL
SELECT * FROM SignupCTE
UNION ALL
SELECT * FROM RideRequestCTE
UNION ALL
SELECT * FROM RidesAcceptedCTE
UNION ALL
SELECT * FROM PickupsCTE
UNION ALL
SELECT * FROM PaymentCTE
UNION ALL
SELECT * FROM ReviewsCTE
ORDER BY step, age_range, funnel_count DESC;
```

-- Requests by Hour of Day
```
SELECT EXTRACT(HOUR FROM request_ts) AS hour_of_day,
    COUNT(*) AS request_count, COUNT(transaction_ts) AS transaction_count
FROM metrocar
WHERE EXTRACT(HOUR FROM request_ts) IS NOT NULL
GROUP BY EXTRACT(HOUR FROM request_ts)
ORDER BY hour_of_day;
```

```
-- Group by Funnel Step, Hours of Day (Ride Granuality)
WITH FunnelCTE AS (
  SELECT
  1 AS step,
        'App Downloads' AS funnel_step,
        EXTRACT(HOUR FROM download_ts) AS hour_of_day,
```

```
    COUNT(download_ts) AS funnel_count
 FROM metrocar
 WHERE EXTRACT(HOUR FROM download_ts) IS NOT NULL
 GROUP BY 1, 2, 3
 ORDER BY 1, 2, 3
),
SignupCTE AS (
 SELECT
 2 AS step,
        'Signups' AS funnel_step,
        EXTRACT(HOUR FROM signup_ts) AS hour_of_day,
    COUNT(signup_ts) AS funnel_count
 FROM metrocar
 WHERE EXTRACT(HOUR FROM signup_ts) IS NOT NULL
        GROUP BY 1, 2, 3
 ORDER BY 1, 2, 3
),
RideRequestCTE AS (
  SELECT
                        3 AS step,
    'Ride Requests' AS Funnel_Step,
     EXTRACT(HOUR FROM request_ts) AS hour_of_day,
    COUNT(request_ts) AS funnel_count
  FROM metrocar
  WHERE request_ts IS NOT NULL
        GROUP BY 1, 2, 3
        ORDER BY 1, 2, 3
),
RidesAcceptedCTE AS (
  SELECT
                    4 as step,
                    'Rides Accepted' AS Funnel_Step,
    EXTRACT(HOUR FROM accept_ts) AS hour_of_day,
    COUNT(accept_ts) AS funnel_count
  FROM metrocar
  WHERE accept_ts IS NOT NULL
        GROUP BY 1, 2, 3
        ORDER BY 1, 2, 3
),
PickupsCTE AS (
  SELECT
                    5 as step,
    'Pickups' AS Funnel_Step,
     EXTRACT(HOUR FROM pickup_ts) AS hour_of_day,
    COUNT(pickup_ts) AS funnel_count
  FROM metrocar
  WHERE pickup_ts IS NOT NULL
        GROUP BY 1, 2, 3
```

Robert Tiger


```sql
        ORDER BY 1, 2, 3
),
PaymentCTE AS (
   SELECT
                        6 as step,
      'Payment' AS Funnel_Step,
       EXTRACT(HOUR FROM transaction_ts) AS hour_of_day,
      COUNT(transaction_ts) AS funnel_count
   FROM metrocar
   WHERE transaction_ts IS NOT NULL
        GROUP BY 1, 2, 3
        ORDER BY 1, 2, 3
)
SELECT
                step,
   Funnel_Step,
   hour_of_day,
   Funnel_Count
FROM FunnelCTE
UNION ALL
SELECT * FROM SignupCTE
UNION ALL
SELECT * FROM RideRequestCTE
UNION ALL
SELECT * FROM RidesAcceptedCTE
UNION ALL
SELECT * FROM PickupsCTE
UNION ALL
SELECT * FROM PaymentCTE
ORDER BY step, hour_of_day;
```

-- All Metrics Grouped
```sql
WITH FunnelStage AS (
 SELECT
   1 AS step,
   'App Downloads' AS funnel_step,
        platform,
        age_range,
   download_ts AS event_ts,
   app_download_key AS user_identifier
 FROM metrocar

 UNION ALL

 SELECT
   2 AS step,
   'Signups' AS funnel_step,
   platform,
```

```
        age_range,
        signup_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE signup_ts IS NOT NULL

UNION ALL

SELECT
  3 AS step,
  'Ride Requests' AS funnel_step,
        platform,
        age_range,
  request_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE request_ts IS NOT NULL

UNION ALL

SELECT
  4 AS step,
  'Rides Accepted' AS funnel_step,
        platform,
        age_range,
  accept_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE accept_ts IS NOT NULL

UNION ALL

SELECT
  5 AS step,
  'Pickups' AS funnel_step,
        platform,
        age_range,
  pickup_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE pickup_ts IS NOT NULL

UNION ALL

SELECT
  6 AS step,
  'Payment' AS funnel_step,
        platform,
```

```
      age_range,
    transaction_ts AS event_ts,
    user_id::text AS user_identifier
  FROM metrocar
  WHERE transaction_ts IS NOT NULL

  UNION ALL

  SELECT
    7 AS step,
    'Left a Review' AS funnel_step,
         platform,
         age_range,
    dropoff_ts AS event_ts,
    user_id::text AS user_identifier
  FROM metrocar
  WHERE review_id IS NOT NULL
),
StageTotalCounts AS (
  SELECT
    step,
    COUNT(DISTINCT user_identifier) AS total_count_for_stage
  FROM FunnelStage
  GROUP BY step
)
SELECT
  f.step,
  f.funnel_step,
  f.platform,
  f.age_range,
  CONCAT(EXTRACT(YEAR FROM f.event_ts), '-', LPAD(EXTRACT(MONTH FROM f.event_ts)::text, 2,
'0')) AS year_month,
  COUNT(DISTINCT f.user_identifier) AS funnel_count,
  s.total_count_for_stage
FROM FunnelStage f
JOIN StageTotalCounts s ON f.step = s.step
GROUP BY f.step, f.funnel_step, f.platform, f.age_range, year_month, s.total_count_for_stage
ORDER BY f.step, f.funnel_step, f.platform, f.age_range, year_month;

-- Ride Level
WITH FunnelStage AS (
  SELECT
    3 AS step,
    'Ride Requests' AS funnel_step,
         platform,
         age_range,
    request_ts AS event_ts,
    user_id::text AS user_identifier
```

Robert Tiger

```sql
FROM metrocar
WHERE request_ts IS NOT NULL

UNION ALL

SELECT
  4 AS step,
  'Rides Accepted' AS funnel_step,
      platform,
      age_range,
  accept_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE accept_ts IS NOT NULL

UNION ALL

SELECT
  5 AS step,
  'Pickups' AS funnel_step,
      platform,
      age_range,
  pickup_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE pickup_ts IS NOT NULL

UNION ALL

SELECT
  6 AS step,
  'Payment' AS funnel_step,
      platform,
      age_range,
  transaction_ts AS event_ts,
  user_id::text AS user_identifier
FROM metrocar
WHERE transaction_ts IS NOT NULL

UNION ALL

SELECT
  7 AS step,
  'Left a Review' AS funnel_step,
      platform,
      age_range,
  dropoff_ts AS event_ts,
  user_id::text AS user_identifier
```

```
  FROM metrocar
  WHERE review_id IS NOT NULL
),
StageTotalCounts AS (
  SELECT
    step,
    COUNT(user_identifier) AS total_count_for_stage
  FROM FunnelStage
  GROUP BY step
)
SELECT
  f.step,
  f.funnel_step,
  f.platform,
  f.age_range,
  CONCAT(EXTRACT(YEAR FROM f.event_ts), '-', LPAD(EXTRACT(MONTH FROM f.event_ts)::text, 2,
'0')) AS year_month,
  COUNT(f.user_identifier) AS funnel_count,
  s.total_count_for_stage
FROM FunnelStage f
JOIN StageTotalCounts s ON f.step = s.step
GROUP BY f.step, f.funnel_step, f.platform, f.age_range, year_month, s.total_count_for_stage
ORDER BY f.step, f.funnel_step, f.platform, f.age_range, year_month;
```