Game-Theoretic Mechanisms for Pricing and Matching in Ride-Sharing Services

Outline

1.Introduction
2. Literature Review
A. Themes Discovered in Review
B. Identification of Gaps
3. Scope and Problem Statement
4. Research Challenges
5. Research
6. Objective
7. Methodology
8. Limitations and Future Work
9. Conclusion
10. Guide Approval Mail Screenshot
11. References

Introduction

- Challenges in Dynamic Pricing & Matching – Ride-sharing platforms like Uber face inefficiencies in dynamic pricing and driver-passenger matching, leading to frequent complaints from both riders and drivers regarding unfair fares and long wait times.

- Game-Theoretic Approach – Traditional pricing models often prioritize platform profit over user satisfaction. A game-theoretic model allows a balance between competitive pricing, driver incentives, and passenger affordability, ensuring a win-win situation for all stakeholders.

- Real-Time Optimization – Unlike static pricing models, which struggle with fluctuating demand, our approach incorporates real-time traffic conditions, supply-demand variations, and multi-modal transport options to enhance efficiency.

- Fair & Sustainable Pricing – The project introduces a fairness-driven pricing mechanism that accounts for congestion, weather, and driver availability, ensuring pricing transparency while optimizing platform revenue.

- Multi-Agent System – The model treats drivers, passengers, and the platform as intelligent agents, making strategic decisions based on Nash equilibrium concepts to create a stable and competitive ride-sharing environment.

- Scalability & Future Applications – The proposed system can be adapted for shared mobility solutions beyond ride-hailing, including autonomous ride-sharing, bike taxis, and last-mile connectivity, making it a future-proof solution for urban mobility.

A. Themes Discovered in Review

1. **Dynamic Pricing Models**

- Most studies focus on dynamic pricing mechanisms to balance supply and
- demand for ride-sharing services.
- Game-theoretic approaches (e.g., Stackelberg games, auction-based pricing) are widely used to optimize profits for platforms while maintaining driver and passenger satisfaction. Pricing strategies often consider external factors
- such as traffic, weather, and time of day.

2. **Matching Algorithms**

- Many papers propose algorithms to optimize driver-passenger matching.
- Matching models incorporate real-time factors such as geographical locations, estimated arrival times, and passenger preferences. The use of Nash equilibrium
- and cooperative game theory is explored to ensure fairness in the matching process.

3. **Driver Incentives**

- Several studies address incentive mechanisms to motivate drivers to move to high-demand areas or accept ride-pooling requests.
- Game-theoretic approaches model driver behavior and competition in high-demand scenarios.

B. Identification of Gaps:

- Limited Use of Nash Equilibrium in Dynamic Scenarios
- While some papers use Nash equilibrium, its application in highly dynamic and uncertain environments (e.g., traffic fluctuations, sudden demand spikes) is limited.
- Lack of Multi-Agent Collaboration Models
- Most studies treat drivers and passengers as isolated agents rather than exploring collaborative or coalition-based approaches.
- There's a gap in analyzing how drivers could collaborate to optimize routes or earnings.
- Focus on Platform Profitability Over Sustainability
- Many studies prioritize platform profits without fully addressing environmental or social sustainability.
- Integrating sustainability metrics into game-theoretic models is underexplored.
- Inadequate Real-Time Feasibility Testing
- Many proposed algorithms and models are tested only in simulated environments.
- Few studies validate their models with real-world datasets or operational scenarios.

Scope and Problem Statement

- **Scope:**

1. Application of Game Theory: Utilize game-theoretic models to enhance the efficiency and fairness of ride-sharing services.
2. Real-Time Dynamic Pricing: Develop an adaptive pricing mechanism that adjusts based on demand, congestion, and driver availability.
3. Multi-Agent Optimization: Optimize interactions between passengers, drivers, and the platform by modeling them as rational decision-makers in a strategic environment.
4. Fairness and Sustainability: Ensure that pricing strategies benefit both drivers and passengers while maintaining platform profitability.

- **Problem Statement:**

1. Dynamic Pricing Inefficiencies: Ride-sharing platforms often impose surge pricing that may be unfair to passengers and fail to incentivize drivers efficiently.
2. Mismatched Driver-Passenger Assignments: Ineffective matching algorithms can result in longer wait times, empty rides, and inefficient use of resources.
3. Traffic Congestion and Environmental Impact: Poorly optimized ride allocation increases congestion, fuel consumption, and carbon emissions in urban areas.

Research Challenges

- Modeling multi-agent interactions between passengers, drivers, and platforms.
- Balancing competing objectives: profit maximization vs. user satisfaction.
- Implementing real-time solutions with scalability.
- Ensuring Fairness in Pricing and Matching – Addressing biases in pricing algorithms to prevent overcharging passengers in high-demand areas and ensuring equitable earnings for drivers.
- Incorporating Real-Time External Factors – Integrating live traffic conditions, weather data, and socio-economic variations into dynamic pricing and ride-matching decisions.
- Handling Uncertainty and Strategic Behavior – Dealing with unpredictable passenger demand, driver availability, and strategic actions like drivers rejecting low-fare rides or passengers manipulating location data.

Research Objective

1 Multi-Agent Interaction Modeling

Ride-sharing involves multiple stakeholders—passengers, drivers, and the platform—each with different objectives. The framework should model:

- Passengers: Minimize fare while ensuring shorter waiting times and faster travel.
- Drivers: Maximize eaxxrnings while minimizing idle time and fuel consumption.

2. Dynamic Pricing Mechanism Using Game Theory

Dynamic pricing adjusts fares in real-time based on supply-demand conditions. The framework should:

- Implement surge pricing strategies using Nash equilibrium, where drivers and passengers make optimal choices.
- Consider utility-based pricing models that balance affordability for passengers and profitability for drivers.

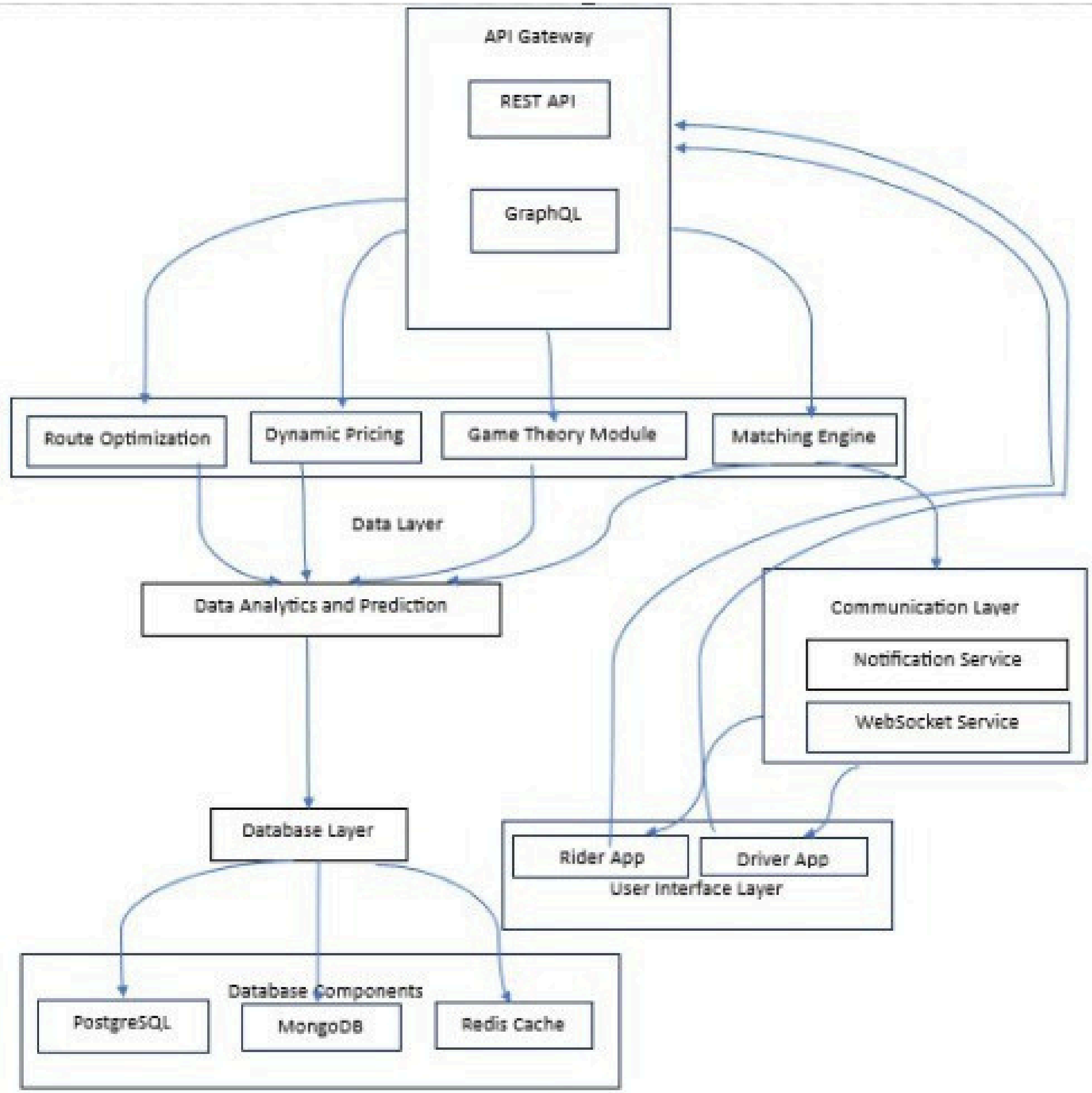
Methodology

1.Centralized Matching with Nash Equilibrium

- Match riders and drivers by optimizing utility for both parties using Nash equilibrium principles, ensuring no participant can improve their outcome by changing their strategy unilaterally.

2.Dynamic Pricing and Route Optimization

- Implement dynamic pricing to balance demand and supply while optimizing routes with real-time traffic data to enhance efficiency and user satisfaction.


API Gateway
REST API
GraphQL
Route Optimization
Dynamic Pricing
Game Theory Module
Matching Engine
Data Layer
Data Analytics and Prediction
Communication Layer
Notification Service
WebSocket Service
Database Layer
Rider App
Driver App
User Interface Layer
Database Components
PostgreSQL
MongoDB
Redis Cache

Smart Ride Tamil Nadu – Detailed Features & Technology

Smart Ride Tamil Nadu is an application designed to serve as a state-of-the-art ride-sharing
platform addressing multiple commuting and service needs of users in Tamil Nadu, combining state-of-the-art technology with deep locality and user-oriented features. Designed on a modern web stack, the Smart Ride Tamil Nadu application delivers high performance, ease of use, and scalability.
The frontend is built with React 18.3.1 and TypeScript, assuring that a strong and maintainable code base is followed. Vite has been used as an alternative to webpack as a build tool and a development server to speed up the build process and hot module reloading, thus improving developer experience.

Tailwind CSS is used for styling, and it adheres to a utility-compatible-first responsible design approach for consistency and speed in UI development. Rich user interface components are built with completely customizable and accessible Radix UI primitives. Interactive maps take central stage in rendering the experience with React-Leaflet and OpenStreetMap integration to visualize real-time pickup locations, drop-off points, and driver locations.

Authentication & User Management

In Smart Ride Tamil Nadu, an authorization scheme in general is assigned to a role, where a driver has a different dashboard and experience from a passenger. Users can register through email, manage their profiles and premium memberships to avail additional features.

A secure Authentication Flow Gives Them Personalized Service

Booking System & Service Types

Consisting of a variety of ride types and service types, the booking system is designed to meet specific user needs. These include Regular rides for everyday travel; Driver-only service for users wishing to hire a driver for their car; Tourist guide drivers to assist travelers in local navigation and language; Luggage helper service for those needing extra careful assistance with bags; Senior care transport services designed for elderly or mobility challenged users.

Users have the option to select pick-up and drop-off location through manual input, or they can use voice-enabled booking, which accepts commands in Indian English, transcribes them in real-time, and provides voice feedback.

Being able to book through speaking commands makes the voice booking assistant a prime application of natural language processing. This gave users the ability to book for their rides via spoken commands, receive voice confirmations, and talk to the system in a hands-free manner.

Driver Features & Dashboard

Drivers have a full-fledged dashboard to monitor their earnings (daily, weekly, monthly), active hours, performance metrics, etc.

Skill-based matching is applied to assign drivers only for those rides concerning their skill sets (language spoken, service type), thus raising the quality of rides and customer satisfaction.

Safety Features

➢ An SOS emergency system is embedded for passengers and drivers.
➢ Verification of the driver includes taking a selfie.
➢ Tracking the real-time location for all ongoing rides.
➢ Provision of women-only drivers for female passengers who feel additional safety and comfort is warranted.

Hop-In System

The Hop-In system feature is the dynamic ride continuity, through which if a driver is continuing for only part of the route, a backup driver is automatically allocated just before the cutoff point.

The ride, therefore, continues without input from the passenger.

This feature monitors the driver's distance limit and triggers real-time transition management and notifications so that all rides to great distances or rural areas can be accomplished with certainty.

Premium Features

➢ Premium users receive a host of benefits, including:
➢ 15 percent discounts on all ride types,
➢ Priority matching of drivers during peak hours,
➢ Premium customer support available 24/7,
➢ Special seasonal offers, working to plump the pockets of loyal users.
➢ They add to rider convenience and lessen wait times with safety and comfort.

Driver Selection & Matching

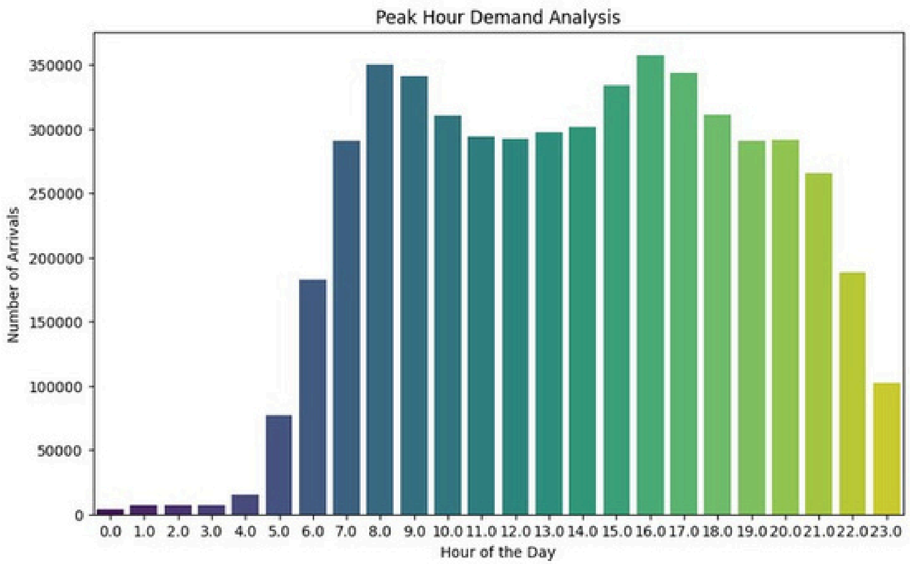
Users can be selective as to whom they want to engage:

- ➢ Skill sets such as local language fluency or city guide experience.
- ➢ Gender preferences.
- ➢ Vehicle type; and
- ➢ A rating system-based selection model where badges give an additional sense of security, and all this personalization only strengthens trust between the users and the platform.

Delhi open transit data analysis:

For game theory we will be using few dataset and delhi transit data is one of those, from these we have gained insights on peak hours in a day so that we can increse the price based on high peak hours and on low peak hours, and congested routes based on this we can take a shortest route.

```python
# Visualize peak hours
plt.figure(figsize=(10,6))
sns.barplot(x=peak_hours.index, y=peak_hours.values, palette='viridis')
plt.title('Peak Hour Demand Analysis')
plt.xlabel('Hour of the Day')
plt.ylabel('Number of Arrivals')
plt.show()
```



Top 10 congested routes:

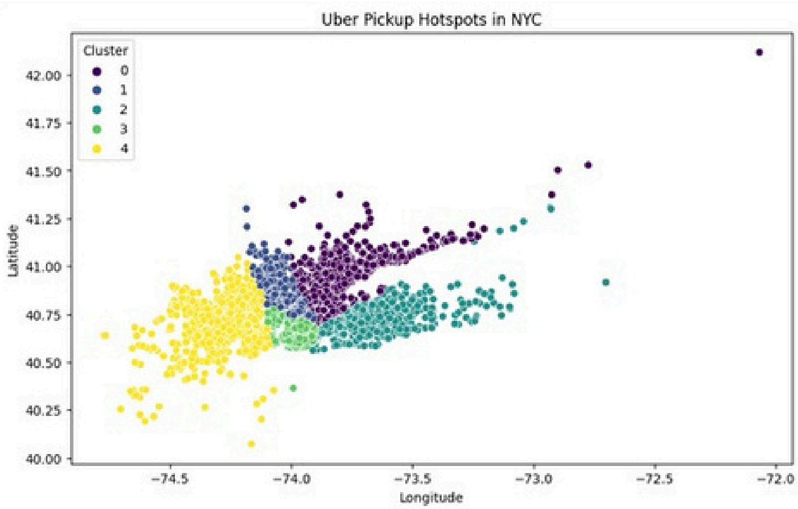
Based on the analysis we can pick a best route with the low congested routes

[11]:
```
# Visualize congestion
plt.figure(figsize=(12,6))
sns.barplot(x=route_congestion.index[:10], y=route_congestion.values[:10], palette='magma')
plt.title('Top 10 Congested Routes')
plt.xlabel('Route ID')
plt.ylabel('Number of Stops')
plt.show()
```

Analysis on uber New york data set:

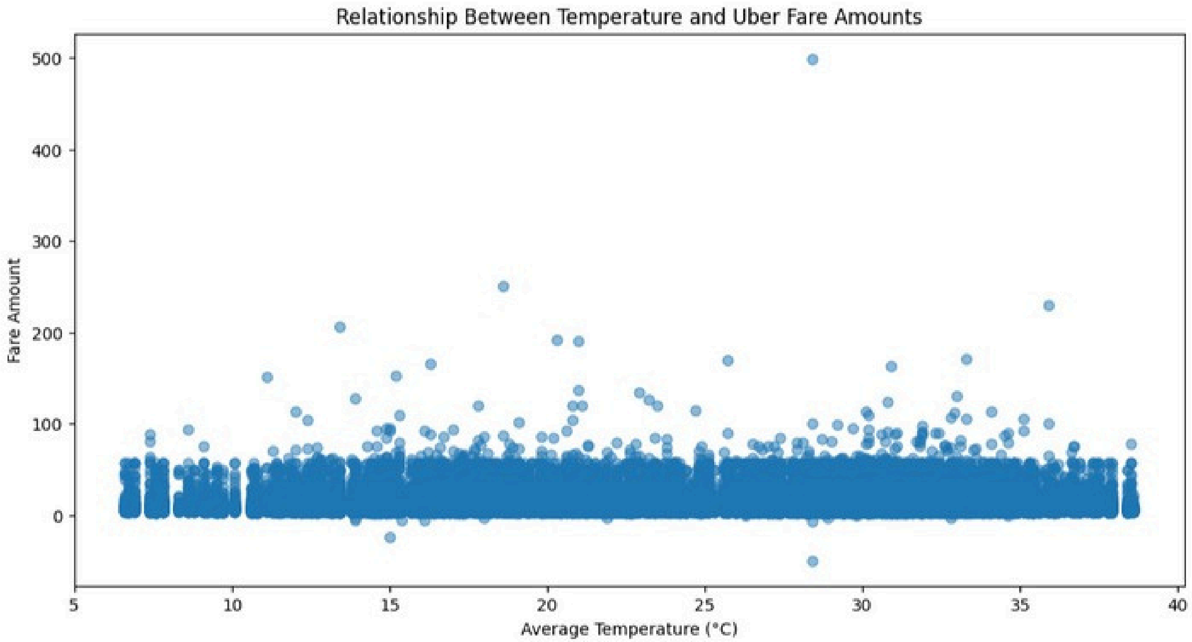
Uber Pickup Hotspots in NYC:
Based on the latitude and on longitude we can pinpoint the location and on that we will be seeing high pickups or passengers surge in daily route we can identify this and based on this we can demand price

```
# Plotting Hotspots
plt.figure(figsize=(10, 6))
sns.scatterplot(x='Lon', y='Lat', hue='Cluster', data=df_cleaned, palette='viridis', legend='full')
plt.title('Uber Pickup Hotspots in NYC')
plt.xlabel('Longitude')
plt.ylabel('Latitude')
plt.show()
```



Weather Data:

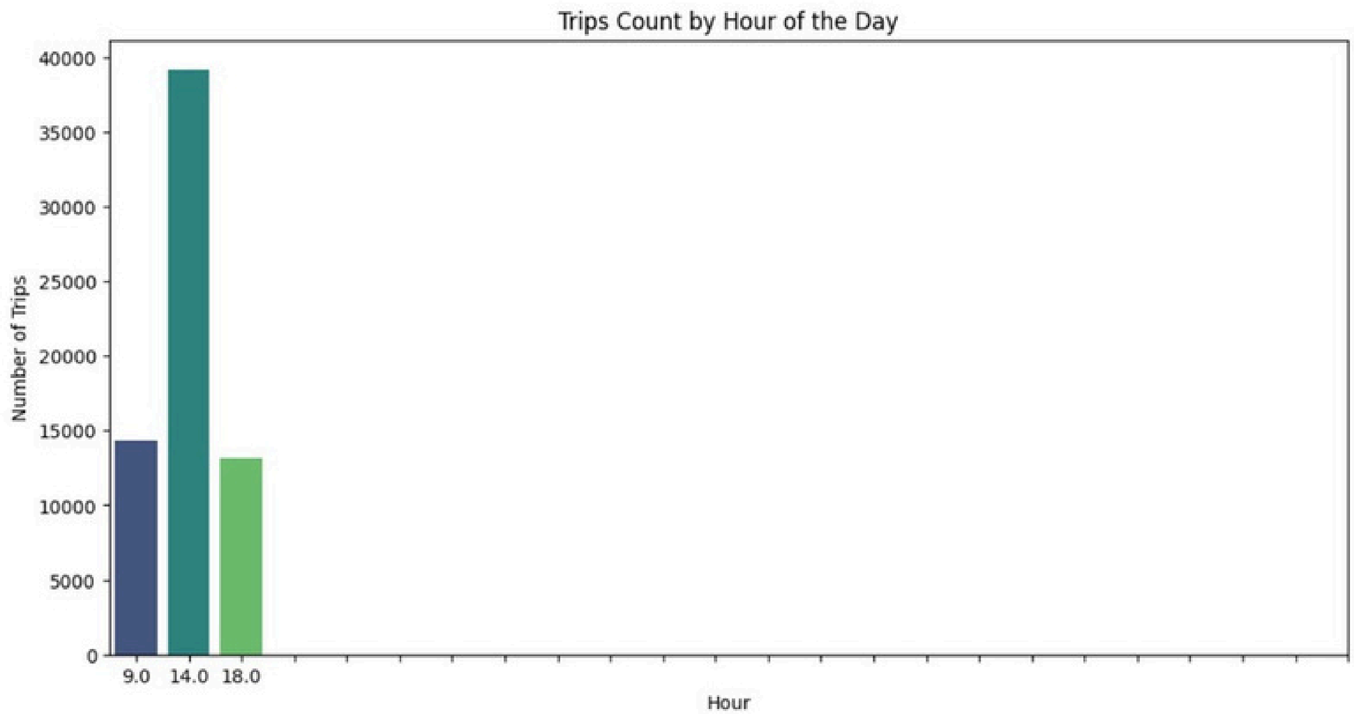
Most fares are clustered below $100, suggesting that the majority of rides are within a typical pricing range.



Peak at 14:00 (2 PM): The highest number of trips (around 40,000) occurs at this time.

Moderate activity at 9:00 AM and 18:00 (6 PM): These hours also show a significant number of trips (~15,000 trips).

No data for other hours: The chart only displays trips for 9 AM, 2 PM, and 6 PM



Nash equlibrium:

We have used Nash equilibrium based on that we will consider few options like weather , traffic, sudden surge of people travelling based on this nash equilibrium will generate price to satisfy driver and raider for dynamic pricing.

```
# Create a bi-matrix game
ride_sharing_game = nash.Game(passenger_payoff, driver_payoff)

# Compute Nash Equilibria
nash_equilibria = ride_sharing_game.support_enumeration()

i = 1
for eq in nash_equilibria:
    print(f"Nash Equilibrium {i}: Passenger Strategy {eq[0]}, Driver Strategy {eq[1]}")
    i += 1
```

```
Nash Equilibrium 1: Passenger Strategy [1. 0.], Driver Strategy [1. 0.]
```

+ Code + Markdown

Smart Ride Tamil Nadu

Sign In

Book Your Ride

Voice commands are not supported in your browser. Please use Chrome, Edge, or Safari for voice features.

Driver-Only Mode

Origin

Select Origin

Destination

Select Destination

Service Type

Regular Ride
Standard ride service

Driver Only
Hire a driver for your vehicle

Tourist Guide Driver
Driver with local tourism knowledge

Luggage Helper
Driver who assists with heavy bags

Senior Care
Patient driver for elderly care

Ride Options

Women Driver Only
Request a female driver for your ride

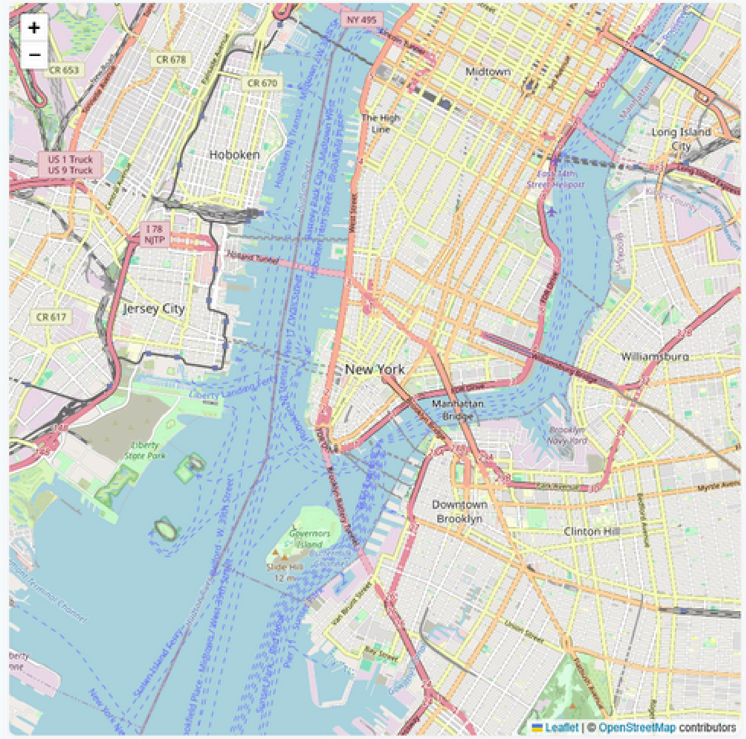
Pay Per Minute
Only pay for the time you use

Enable Ride Sharing (30% fare reduction)

Allow Delivery Pickups (10% additional discount)

Current Surge Factor: 1.08x

Sign in to Book



☆ Upgrade to Premium

Monthly Plan

₹1,499/month

- 15% off on all rides
- Priority driver matching
- 24/7 premium support

Subscribe Monthly

Yearly Plan

Best Value

₹14,999/year

Save ₹2,989 yearly!

- 15% off on all rides
- Priority driver matching
- 24/7 premium support
- Exclusive seasonal offers

Subscribe Yearly

Smart Ride Tamil Nadu

Upgrade to Premium | Sign Out

Driver Status

Available for Rides

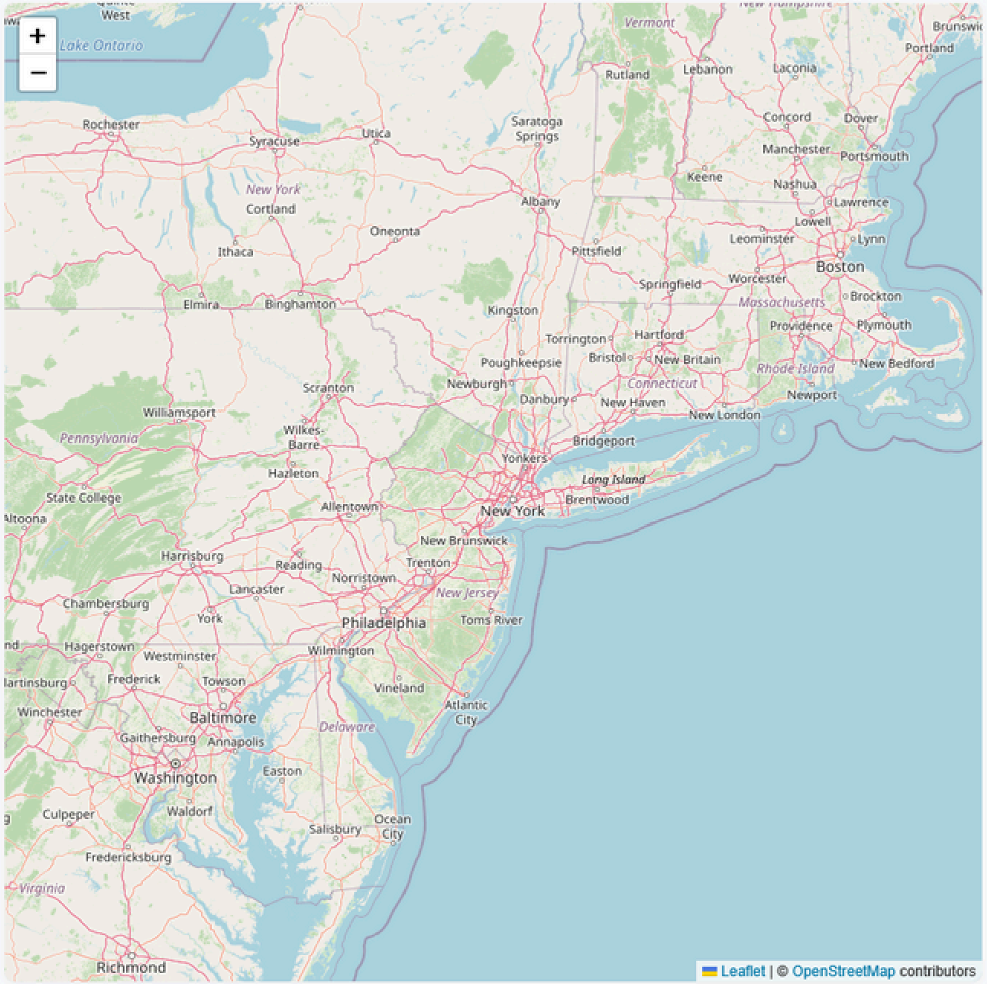
Today's Earnings	Weekly Earnings	Monthly Earnings
₹ **2500**	₹ **15000**	₹ **60000**

Performance Statistics

Total Trips	Rating	Active Hours
1250	4.8	8 hrs

Quick Actions

View Trip History | Update Profile



Limitations and Future Work

- Limitations:

- Dependence on accurate real-time data.
- Computational complexity of large-scale models.

- Future Work:

- Integration with machine learning for predictive capabilities.
- Testing in diverse urban environments.

Conclusion

- Game-theoretic models provide a promising approach for optimizing ride-sharing platforms.
- Balancing platform profits and user satisfaction through efficient pricing and matching mechanisms is achievable.