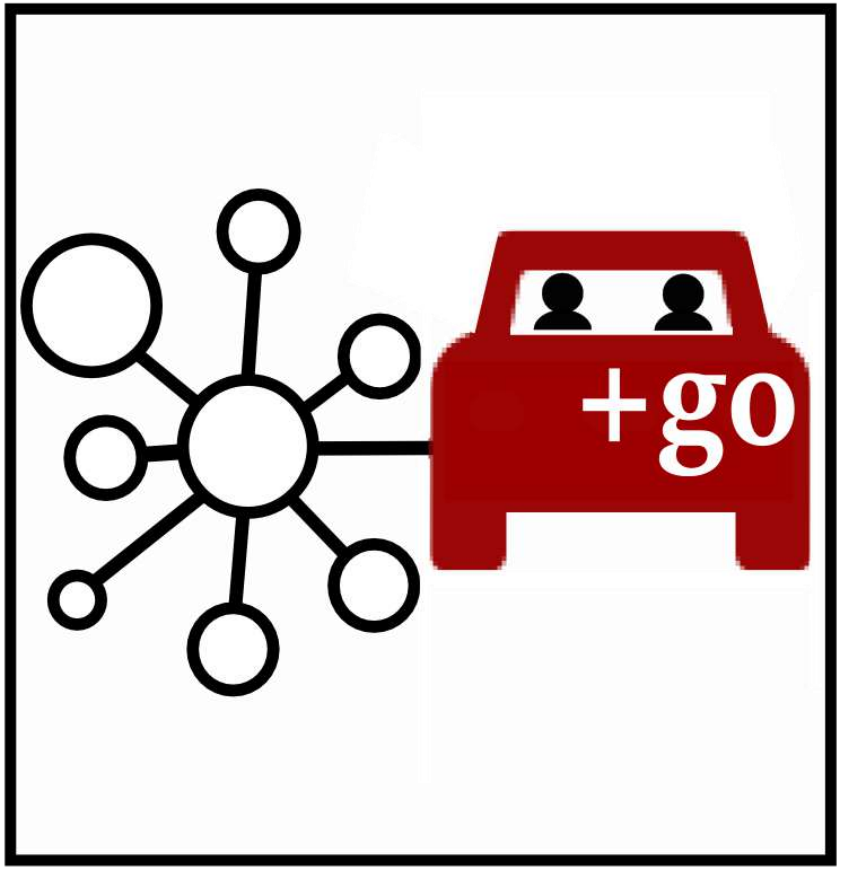
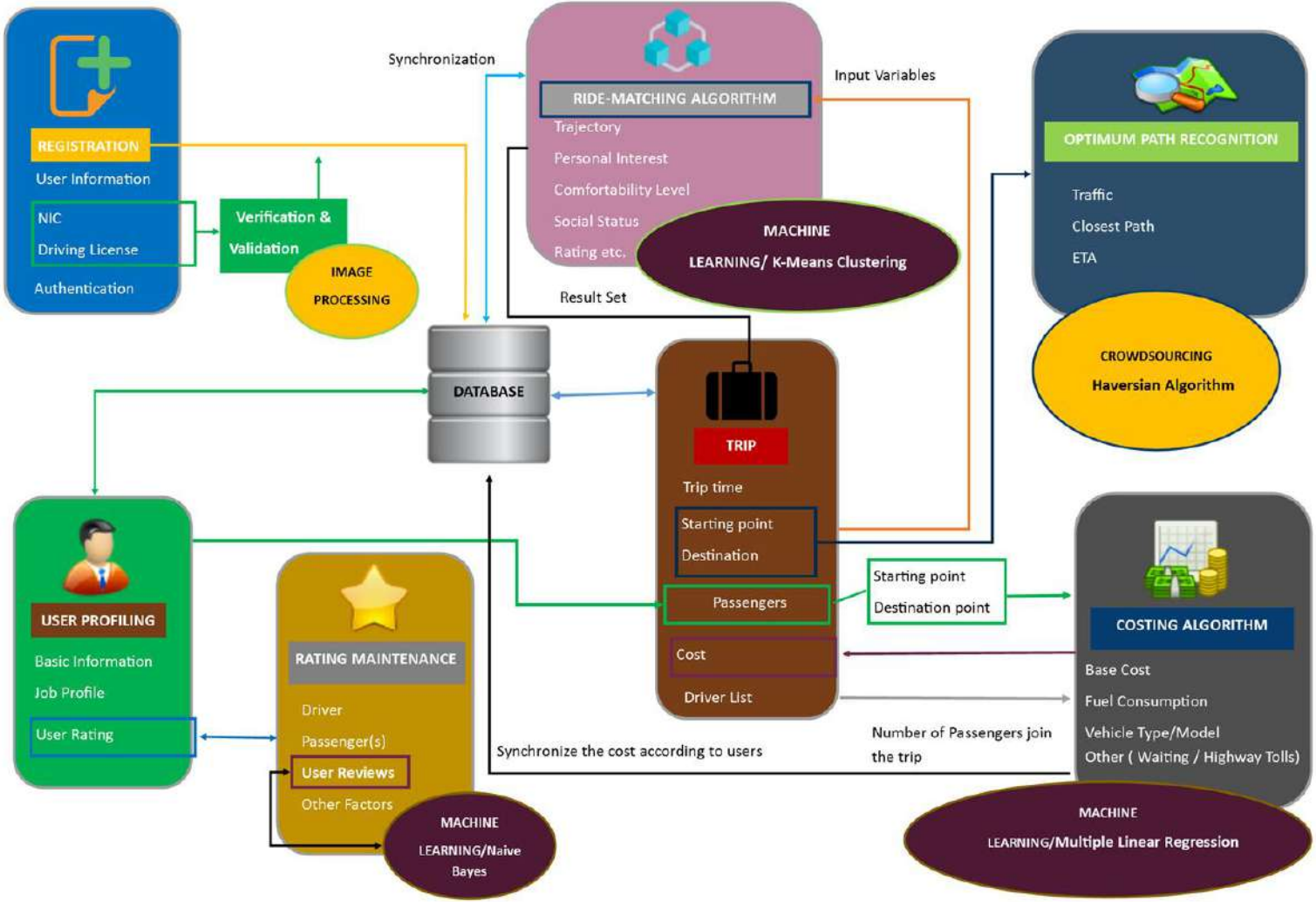
# INTELLIGENT COMPLEMENTARY RIDE-SHARING SYSTEM



# High Level System Diagram


REGISTRATION
User Information
NIC
Driving License
Authentication
Verification & Validation
IMAGE PROCESSING
Synchronization
RIDE-MATCHING ALGORITHM
Trajectory
Personal Interest
Comfortability Level
Social Status
Rating etc.
MACHINE LEARNING/ K-Means Clustering
Input Variables
Result Set
OPTIMUM PATH RECOGNITION
Traffic
Closest Path
ETA
CROWDSOURCING
Haversian Algorithm
DATABASE
TRIP
Trip time
Starting point
Destination
Passengers
Cost
Driver List
Starting point
Destination point
USER PROFILING
Basic Information
Job Profile
User Rating
RATING MAINTENANCE
Driver
Passenger(s)
User Reviews
Other Factors
MACHINE LEARNING/Naive Bayes
Synchronize the cost according to users
Number of Passengers join the trip
COSTING ALGORITHM
Base Cost
Fuel Consumption
Vehicle Type/Model
Other ( Waiting / Highway Tolls)
MACHINE LEARNING/Multiple Linear Regression

# Research Components Identified

- ★ User Profile Management
  - User profiling based on **users' trajectory, driver ratings, social status of driver, time and date, interests of passenger etc.**
  - Document validation and profile rating maintenance
- ★ Optimum path recognition
- ★ Price Calculation

# User Profiling

# Introduction

User Profiling will be based several factors

- Starting point
- Destination point
- Time and date
- User rating
- Type of car
- Available free slots
- Comfortability level
- Waiting time
- Distance to starting point
- Gender preference(People who biased to travel with particular gender)

# Introduction Cont...

## User Profiling will be based several factors continues...

Additionally, Primary rule based on Gender classification will be applied. For example,

If a female leecher or a female driver wants only females to join the journey,

Profiling will include above factors plus additional gender classification factor.

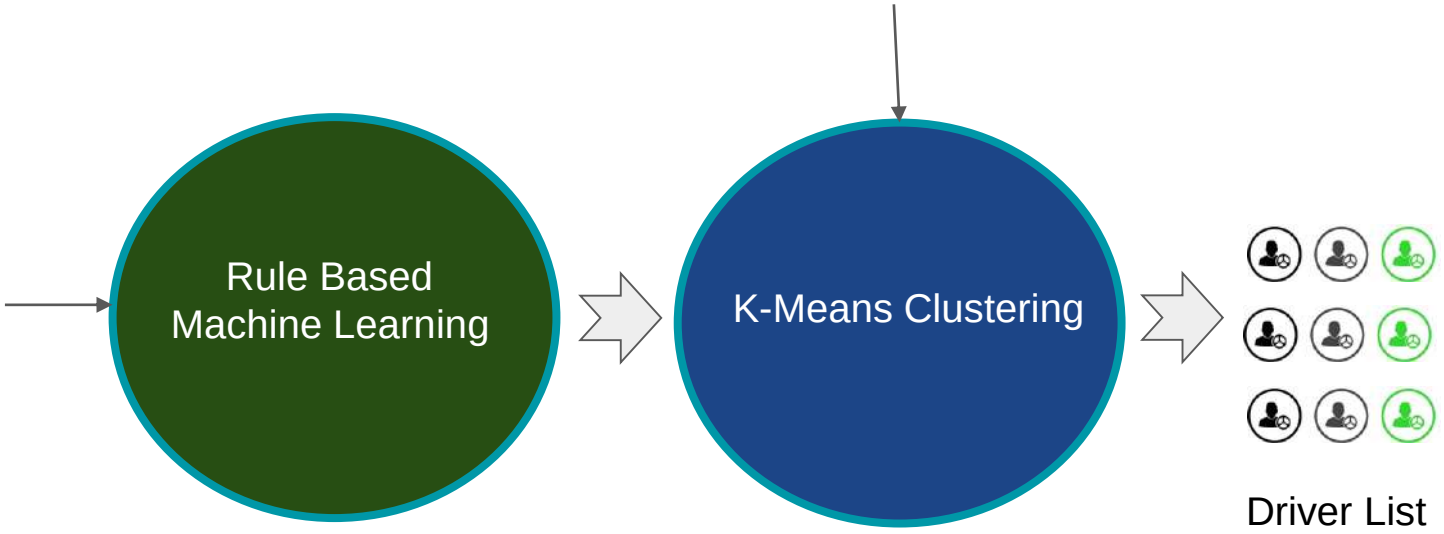
# User Profiling Cont...

To Cluster the users, we are using 2 algorithms.

1. Rule based Machine Learning
2. K-Means Clustering

1. Profession
2. Rating of the driver

1. Trajectory Details
2. Time and Date
3. Car Type
4. Free Slots(>0)
5. Social Status
6. Gender preference



# Methodology- High Level Diagram/ LLD/ Collection of data

Passenger Search For a journey → Ridematching algorithm executes → Display the relevant driver list

- Passenger Should be a **registered** user
- Journey can be searched using several factors
  - **- Trajectory with time**
  - **- Trajectory with vehicle type**

- Retrieve user data of Passenger

**Passenger's Profession/ Interests/ Social Status etc.**
Profession is taken from Details provided in **registration**

**Inputs to the algorithms**

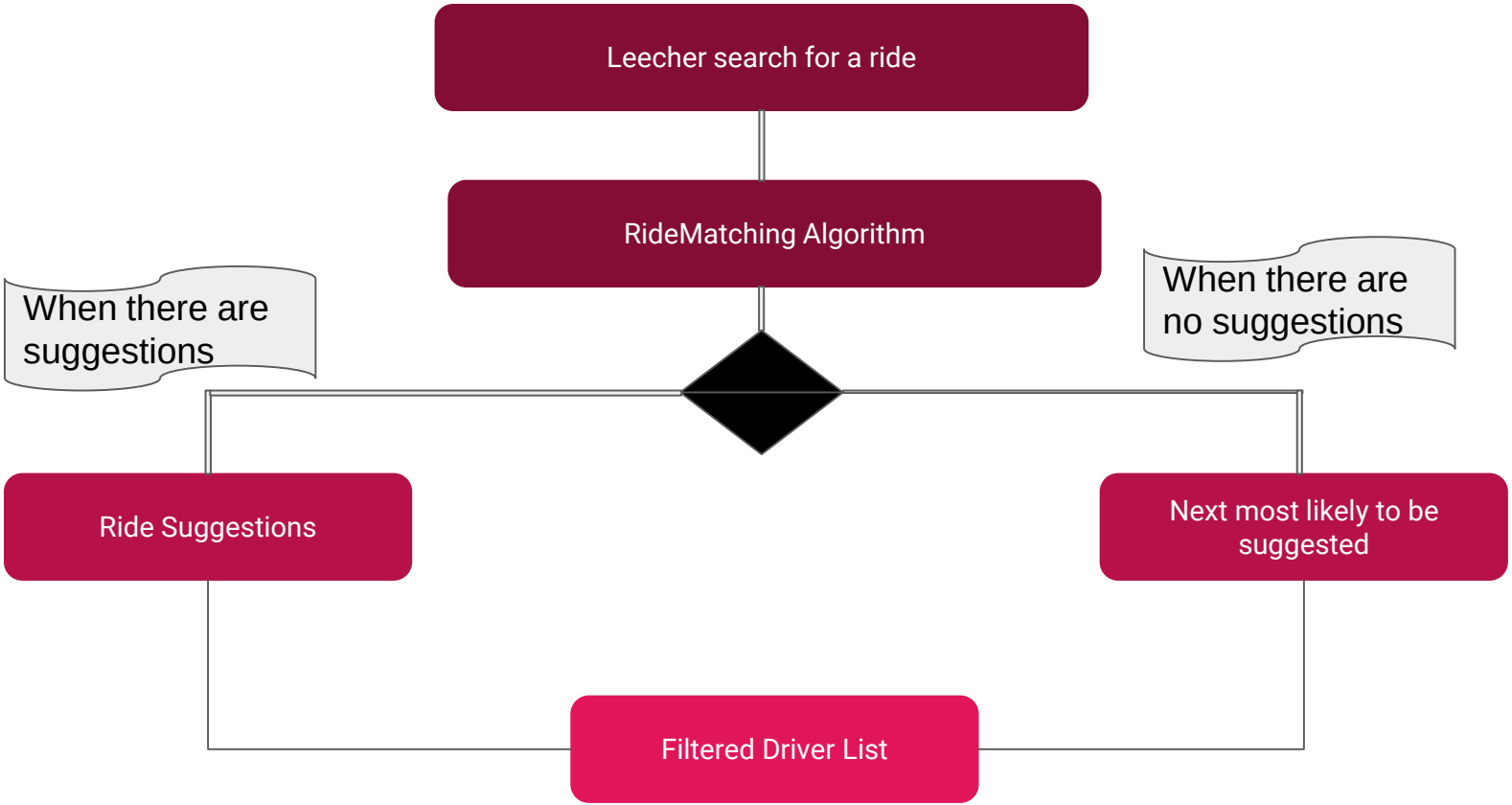
- Trajectory
- Rating
- Time Interval
- Gender preference etc.

Display resultset in order of the Trajectory, Rating etc.

If no exact matching list, display next most suitable matching list.

**Use of Rule based ML and K-Means clustering for easy classification**

# Methodology- Flow Diagram



# Research Goals

- User Profiling based on identified factors(Peer Matching)
- Use the ride-matching algorithm to filter out the drivers from a driver pool
- Build a relationship between same peer groups
- Use of Primary rule for a gender classification
- Get the most related driver list to passengers

# Concepts/ Technologies

- Machine Learning - K-Means Clustering
- Rule based Machine Learning
- Google API/LinkedIn API

# Results

- Identify the drivers who match with passengers itinerary
- Reduction of manual search
- Automatic identification of suitable driver
- Provide the best service to the leechers
- Passenger oriented driver list

# References


- https://dzone.com/articles/10-interesting-use-cases-for-the-k-means-algorithm
- https://mapr.com/blog/monitoring-real-time-uber-data-using-spark-machine-learning-streaming-and-kafka-api-part-1/
- https://ieeexplore.ieee.org/document/7424025

# Document Validation

# Introduction

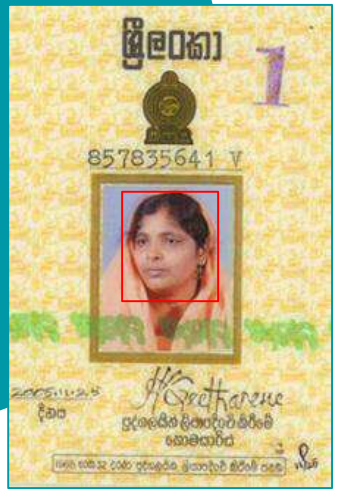
Validation of license,NIC,and their expirations will be identified using image processing algorithm which will be highly efficient to minimize the risk of fake profiles getting registered in the system.

# Methodology- High Level Diagram/ LLD/ Collection of data (Document validation)

- Here we will identify the mandatory elements (user image,nic) that should be included in the driving licence and check whether those elements are included in the image send by the user.
- We will also validate the expiration date of the license.
- Finally we will check the compatibility of the license with the NIC, by checking the NIC number of the image sent of the NIC
- We will also implement the feature for both new and old NIC and licenses.

# Methodology- High Level Diagram/ LLD/ Collection of data (Document validation)

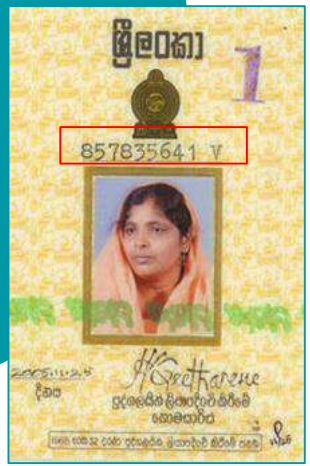
**Step 1**


**Identify whether the images contains a photograph of people**

# Methodology- High Level Diagram/ LLD/ Collection of data (Document validation)

**Step 2**


**Identify whether the images contains NIC and expiration date of license**

# Methodology- High Level Diagram/ LLD/ Collection of data (Document validation)

**Step 3**



**Verify whether the two NIC numbers in id card and license get matched.**

# Research Goals

- Identify the image boundaries of NIC and licence (Should identify both old and newly produced document properties)
- Identify the most important components should be included in each document.
- Identify the expiration dates of the documents and notify the users to make them up to date.

# Concepts/ Technologies

- Image processing
  - OpenCV - To process the image
  - Tesseract - To read text from processed images

# Results

- User validation and verification
- Elimination of manual validation strategies
- Reduction of fake profile registrations
- Notifications about the expiration dates of the licence provided

# References

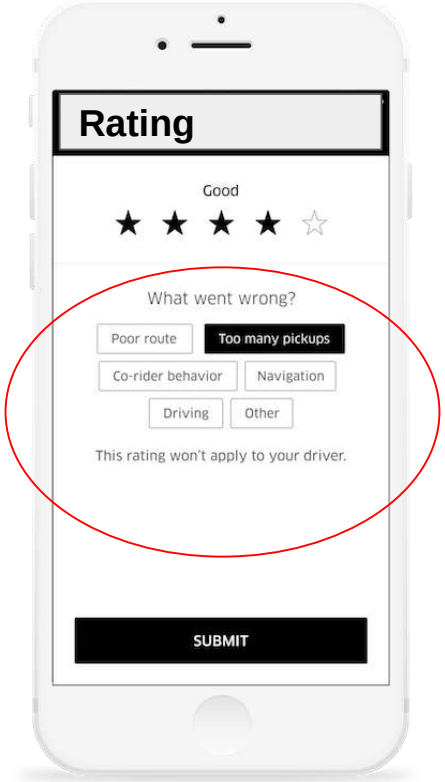

- https://www.leadtools.com/sdk/forms/drivers-license
- https://blog.francium.tech/information-extraction-from-id-cards-using-yolov2-e3c846cb3796
- https://pythonprogramming.net/thresholding-image-analysis-python-opencv-tutorial/

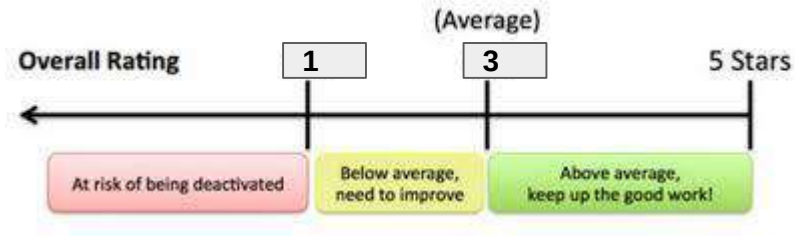
# Profile Rating Maintenance.

# Introduction

Profile rating will be identified at the end of each ride, it will be used to identify the nature of the driver and passengers.Based on the rating, users will be sorted more.

# Methodology - High Level Diagram / LLD / Collection of data (Rating maintenance)



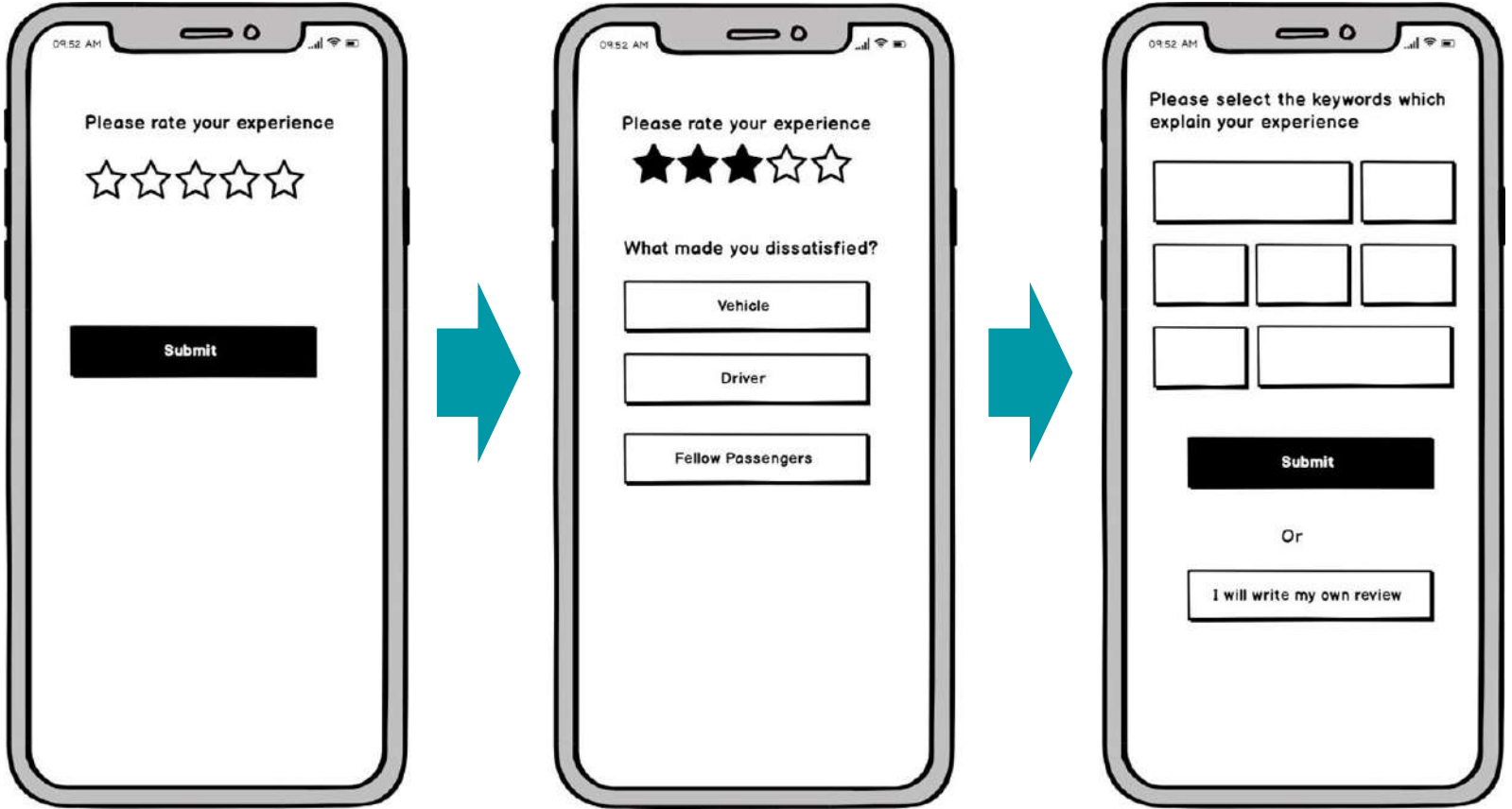

Vehicle | Driver | Fellow passengers | Own reviews

Select passenger

**Sentiment Analysis**

Identify the experience using the keywords

# Methodology- High Level Diagram/ LLD/ Collection of data (Rating maintenance)



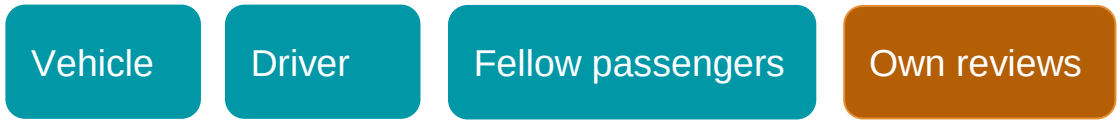
# Methodology- High Level Diagram/ LLD/ Collection of data (Rating maintenance)

We will be using the standard approach like 1 to 5 star rating to rate the overall journey.
Whenever the rating gets lower than 5, we will ask whether

1) Vehicle
2) The driver
3) The fellow passengers

made the rating low.We will provide set of keywords and according to the selections we will understand the user experience.

eg.




**Further we will let the users to write their own reviews (if any). We will proceed with sentiment analysis to identify the experience of the user**

**In cases like harassments, we will give a negative rating for them and they have the risk of getting banned/deactivated from the application**

# Methodology- High Level Diagram/ LLD/ Collection of data (Rating maintenance)

- Default value for the rating will be **5**
- It will be reduced or modified according to the average ratings
- Every user of the system will maintain two ratings in their profile.
    - **Personal behavior**
    - **Vehicle**
- After end of each journey,driver will get to rate passengers (if any) ;default will be 5
- At the times the passengers get down,they get to do their rating. When the rating given by them is below 5, they get to choose which made them low rated.So after it is specified ,all the other categories will get the default value of 5.
- When the users write their own reviews, a rating will be calculated based on the review.The final rating for the trip will be given as an average of **the rating specified by the user** and the **rating identified by the review**

# Research Goals

Identify the response of the passengers and the driver regarding the other passengers joined in the journey and rate the people accordingly,this will also identify the behaviours of the users as well as vehicle conditions

# Concepts/ Technologies

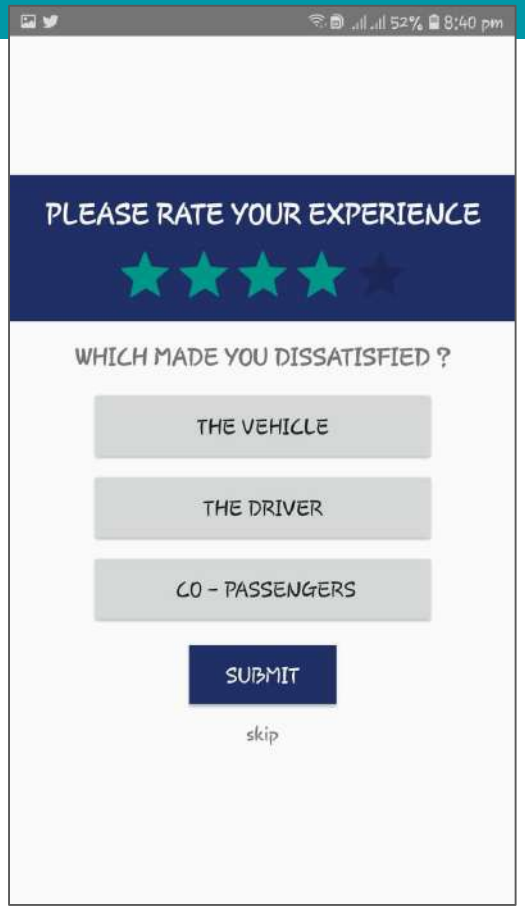
- Basic keyword identification
- Machine learning - Sentiment analysis (**Naive Bayes**)

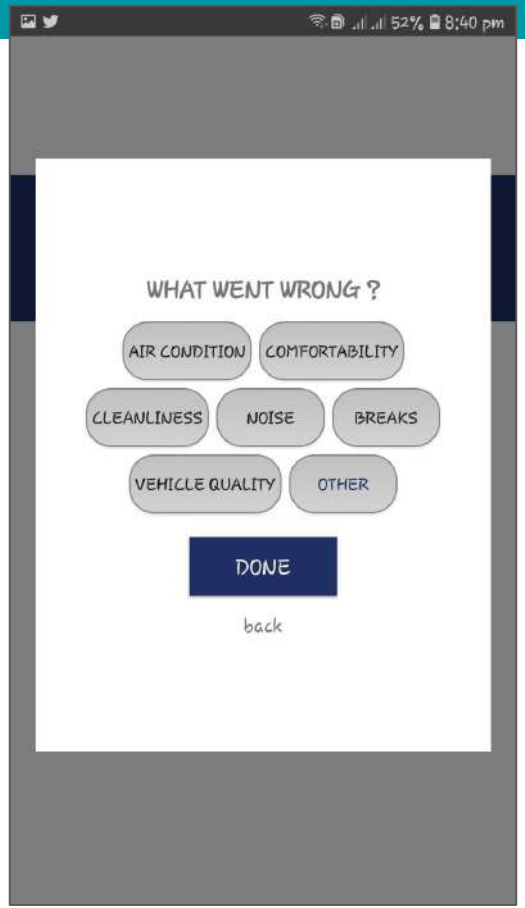
# Results

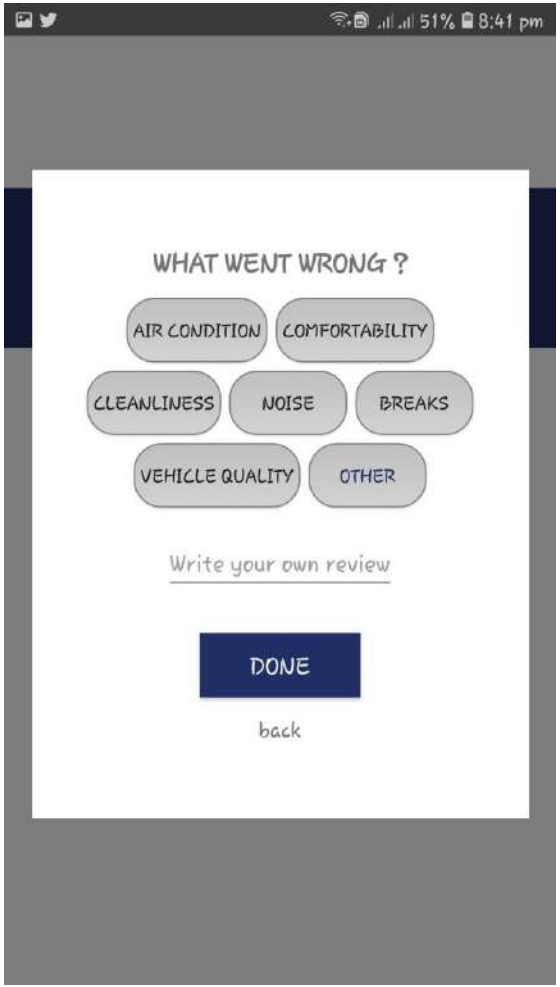
- Ensure the safety and experience of users by analysing reviews
- Provide a better service by considering the ride sharing experience of the users

# Interfaces


52% 8:40 pm
PLEASE RATE YOUR EXPERIENCE
SUBMIT
skip


52% 8:40 pm
PLEASE RATE YOUR EXPERIENCE
WHICH MADE YOU DISSATISFIED ?
THE VEHICLE
THE DRIVER
CO - PASSENGERS
SUBMIT
skip


52% 8:40 pm
WHAT WENT WRONG ?
AIR CONDITION
COMFORTABILITY
CLEANLINESS
NOISE
BREAKS
VEHICLE QUALITY
OTHER
DONE
back


51% 8:41 pm
WHAT WENT WRONG ?
AIR CONDITION
COMFORTABILITY
CLEANLINESS
NOISE
BREAKS
VEHICLE QUALITY
OTHER
Write your own review
DONE
back


51% 8:41 pm
WHAT WENT WRONG ?
BAD NAVIGATION
PROFESSIONALISM
CLEANLINESS
SERVICE
MUSIC
RECKLESS DRIVING
OTHER
DONE
back

# References

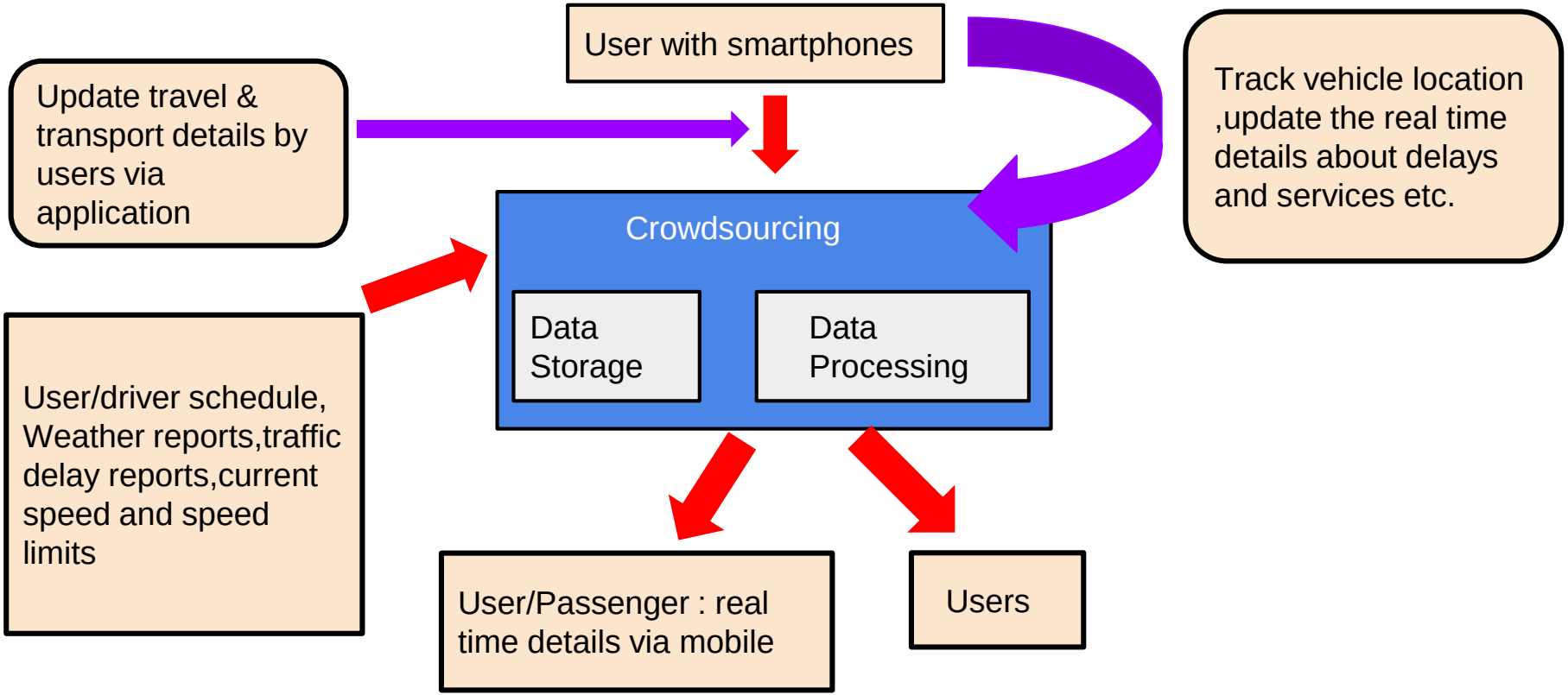

- https://baymard.com/blog/user-perception-of-product-ratings
- https://www.datacamp.com/community/tutorials/simplifying-sentiment-analysis-python
- https://towardsdatascience.com/sentiment-analysis-with-python-part-1-5ce197074184
- https://www.youtube.com/watch?v=O_B7XLfx0ic
- https://monkeylearn.com/sentiment-analysis/
- https://www.kdnuggets.com/2018/03/5-things-sentiment-analysis-classification.html
- https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/

# Optimum path recognition

# Introduction

- In Order to display the route,finding the shortest path which connect starting point and destination ,while tracking the position and order of intermediary locations by using crowdsourcing technology.

# Methodology- High Level Diagram/ LLD/ Collection of data



# Research Goals

- Identify the path with least traffic and the shortest distance to the destination by using haversine algorithm.
- Enable registered users to facilitate to enter live updates(about obstacles,delays,services) on the relevant path.
- Allows displaying a custom map, trip and routing information.
- Show the weather reports,traffic delay reports and schedules to the users.
- Show the estimated time arrival.

# Results

- With the involvement of crowdsourcing vehicle information can be accessed easily where citywide vehicles are inter-connected as an intelligent transportation system.

# Concepts/ Technologies

- CrowdSourcing
- Machine Learning

# References


- https://www.earthdatascience.org/courses/earth-analytics-python/spatial-data-vector-shapefiles/python-customize-map-legends-geopandas/
- https://www.quora.com/How-does-the-algorithm-of-Google-Maps-work
- https://github.com/googlemaps/google-maps-services-python
- https://andrew.hedges.name/experiments/haversine/
- https://blog.goodaudience.com/google-maps-in-python-part-2-393f96196eaf
- https://player.fm/series/technologyiq/my-project-travel-time-bot-with-google-maps-api-and-python-part-2-making-the-travel-data-useful
- https://easternpeak.com/blog/how-to-develop-a-taxi-booking-app-like-uber/

# Price Calculation

# Introduction

- Price Calculation is basically calculate via Machine Learning (Multiple Linear Regression) concept that will be helpful decide reasonable price to passengers for their journey.
- Full amount of the journey shares among the passengers and driver by specific ratio.

# Methodology- High Level Diagram/ LLD/ Collection of data

- End of the journey , application shows the price of ride to the Customer.That price not should be fixed rate it will differ according to the some specific criterias.

**Base Cost**

- ➢ Reservation Cost and Tax will include the base cost.
- ➢ If Demand is increasing Reservation cost will be increasing (According to number of request receiving the server) .If Request exceed more than standard Reservation cost Multiply by X Percentage.

# Methodology - High Level Diagram/ LLD/ Collection of data

## Cost for Fuel Consumption

Fuel Consumption is calculating according to the Vehicle model, Transmission Type, Fuel Type, Manufacture year and engine Capacity.
Vehicle details are collecting from the survey.

- **Passengers**

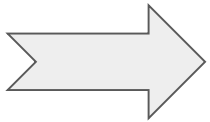
Total Number of Passengers join in for Session.
Driver also consider as a Passenger.
Session Cost will be divided among that number of Passengers, that proportion will added their final cost.

- **Waiting Cost**

If a Leecher is late for more than two minutes or more the cost will be increased, then that amount added to final Cost of who is late.

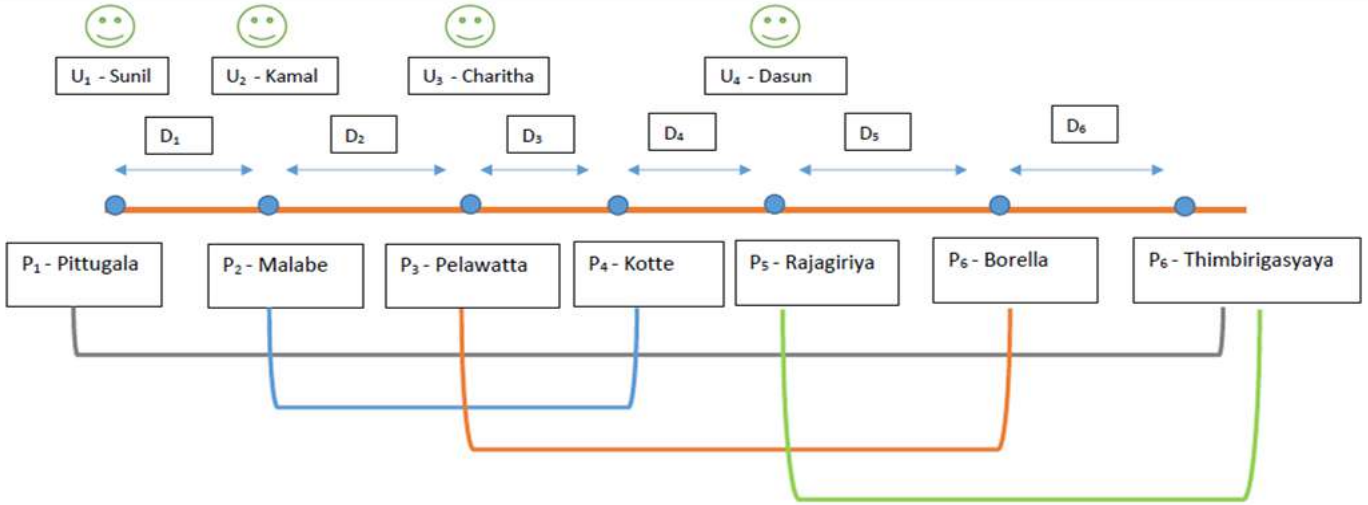
# Methodology - High Level Diagram/ LLD/ Collection of data

- Fuel Consumption may differ according to the **vehicle models**.
- Therefore I have conducted a survey regarding the **fuel consumption details.**
- The aim of the survey was to collect a real data set and **train that data set** using Multiple Linear regression in machine Learning and **predict fuel consumption** according to the vehicle brand ,engine capacity , fuel transmission , fuel type etc.



Here is the Survey

# Methodology - High Level Diagram/ LLD/ Collection of data




$D_1$ - distance between pittugala and Malabe
$D_2$ - distance between malabe and Pelawatta
$D_3$ - distance between Pelawatta and Kotte
f - Cost for fuel consumption per km
L - Number of Current Leechers
C - Total Cost of the Ride
$C_1$ - Cost for $D_1$

Let's say Sunil get into the car from pittugala and Kamal pick up from Malabe and Charitha join the journey from pelawatta. Kamal's office situated near kotte then he get off the journey.one seat available due to Kamal's drop off .Dasun pick up from Rajagiriya and went till the end of the journey

$$C\ i\ (\text{Cost for Di}) = \frac{D_i \times f}{\sum L}$$

$$\text{Total Cost of the Ride} = \sum_{i=Start\ point}^{end\ point} (\text{C i}) + Base\ Cost + Waiting\ Cost + Additional\ Cost\ (Highway\ tolls)$$

# Methodology - High Level Diagram/ LLD/ Collection of data

**Cost of User2 ($U_2$) = Cost for between Malabe and Pelawatta ($C_1$) + Cost for between Pelawatta and Kotte ($C_2$) + Base + Waiting Cost + Additional Cost (Highway tolls / Parking)**

Cost of User2 ($U_2$) = $C_1 + C_2$ + Base + Waiting

Cost for between Malabe and Pelawatta ($C_2$) = $$\frac{D_2\ (Distance\ between\ Malabe\ and\ Pelawatta)\ x\ f}{3\ (No\ of\ Current\ Leechers)}$$

Cost for between Pelawatta and Kotte ($C_3$) = $$\frac{D_3\ (Distance\ between\ Pelawatta\ and\ Kotte)\ x\ f}{4\ (No\ of\ Current\ Leechers)}$$

Cost of User 2 ($U_2$) = $\frac{D_2 x f}{3} + \frac{D_3 x f}{4} + Base + Waiting\ Cost + Additional\ Cost\ (Highway\ tolls)$

# Research Goals

- Find the factors affecting to the price of the journey.
- Find out the proportion of the total price need to pay by each leecher in the journey
- Consider approaches to be used when on mondays/fridays where people may need to come through expressways (eg.expressway cost)

# Concepts/ Technologies

- Machine Learning (Multiple Linear Regression)
- Google Map API

# Results

- Cost Calculation Equation
- Cost Calculation Convert into the Algorithm
- Predict the fuel consumption according to the vehicle
- Display Estimate Cost of the ride

# References


- The Cost of Convenience: Ridesharing and Traffic Fatalities John M. Barrios, Yael V. Hochberg, and Hanyi Livia Yi
- Real Real-Time Carpooling and Ride-Sharing: Position paper on Design Concepts, Distribution and Cloud Computing Strategies
- Cost-based analysis of autonomous mobility services