MACHINE LEARNING-BASED VEHICLE PERFORMANCE ANALYZER

NALAIYA THIRAN PROJECT BASED LEARNING

On

PROFESSIONAL READINESS FOR INNOVATION, EMPLOYABILITY, AND ENTREPRENEURSHIP

A PROJECT REPORT

TEAM ID: PNT2022TMID46974

Team members:

JOSHE RAJ S P	-	210819104043
KARTHIKEYAN S	-	210819104048
KARAN B	-	210819104046
JAGAN U	-	210819104036

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE ENGINEERING

KINGS ENGINEERING COLLEGE - IRUNGATTUKOTTAI

(A Constituent College of Anna University, Chennai)

IRUNGATTUKOTTAI – 602117

NOVEMBER 2022

TABLE OF CONTENTS

INTRODUCTION

1.1 Project Overview

Forest fires are a major environmental issue, creating economic and ecological damage while endangering human lives. There are typically about 100,000 wildfires in the United States every year. Over 9 million acres of land have been destroyed due to treacherous wildfires. It is difficult to predict and detect Forest Fire in a sparsely populated forest area and it is more difficult if the prediction is done using ground-based methods like Camera or Video-Based approach. Satellites can be an important source of data prior to and also during the Fire due to its reliability and efficiency. The various real-time forest fire detection and prediction approaches, with the goal of informing the local fire authorities.

Automotive Technologies are providing improvised services to the driver's safety and vehicle security under the umbrella of Intelligent Transportation System (ITS). In the development of ITS, advanced Automotive Technologies shall play a crucial role in determining the overall experience of users by making it much at ease in terms of reducing the risk of road accidents, risk of cybercrime in the vehicle, buying a used car etc. It is often noted that judging the driver's driving skill is subjective and is difficult to set a standard for driver's skills. The modern approach to transportation system is focusing on rapidly evolving with the intelligent vehicles. High rise in recorded traffic density, road accidents and crisis faced in regulating the effective management of traffic control in urban and rural 4 areas have concerned us to develop a smart solution in context to ITS.

The automotive industry has great expectations from these futuristic solutions to improve the safety of people and security of vehicles. It is observed that the users are shifting from individualistic approach to the data-centric approach based on OBD-II scanner to avail the augmented driving experience. In spite of the modern command, control, communication, computers and intelligent systems, we are still facing numerous calamities in which thousands of precious humans lives are lost in accidents. Therefore, it should be an immediate need to tackle the small scale yet serious issues using the state-of-the-art techniques. We are mainly focusing on analyzing the data which is collected from the vehicle using the OBD-II scanner and eventually providing the driver's safety solutions. We aim to obtain the solutions by

observing the blind-spots accurately and efficiently using pattern recognition techniques from supervised learning.

2. LITERATURE SURVEY

- Singh D, Singh M., "Internet of Vehicles for Smart and Safe Driving", International Conference on Connected Vehicles and Expo (ICCVE), Shenzhen, 19 -23 Oct.,2015. (This paper has discussed about smart transportation services in cloud (Cloud-STS) for safety and convenience. STS provide driver centric board services in the cloud networks.

- STS composed of Vehicle to Wi Fi networks (V to Wi Fi), Vehicle to Cloud Network (V to CN), Vehicle to Vehicle (V to V), and Cloud Network to service provider (CN to SP). The idea is to utilize the (Wi Fi enabled) Smart Highways and 3Dcameraenabled dash board 5 navigation device to enhance accident prevention /monitoring and control.)

- Zhang, Y., Lin, W., and Chin, Y., "Data -Driven Driving Skill Characterization: Algorithm Comparison and Decision Fusion," SAE Technical Paper2009 -01 -1286, 2009, https://doi.org/10.4271/2009 -01 -1286.Azevedo, C. L Cardoso. (By adapting vehicle control systems to the skill level of the driver, the overall vehicle active safety provided to the driver can be further enhanced for the existing active vehicle controls, such as ABS, Traction Control, Vehicle Stability Enhancement Systems.

- As a follow-up to the feasibility study in, this paper provides some recent results on data-driven driving skill characterization. In particular, the paper presents an enhancement of discriminant features, the comparison of three different learning algorithms for recognizer design, and the performance enhancement with decision fusion. The paper concludes with the discussions of the experimental results and some of the future work.)

- J. E. Meseguer, C. T. Calafate, J. C. Cano and P. Manzoni, "Driving Styles: A smartphone application to assess driver behavior," 2013 IEEE Symposium on Computers and Communications (ISCC), Split,2013, pp.000535 -000540. oi:10.1109/ISCC.2013.6755001. (The Driving Styles architecture integrates both data mining techniques and neural networks to generate a classification of driving styles by analyzing the driver behavior along each route.

- In particular, based on parameters such as speed, acceleration, and revolutions per minute of the engine (rpm), we have implemented a neural network-based algorithm that is able to characterize the type of road on which the vehicle is moving, as well as the degree of aggressiveness of each 6 driver.

- The final goal is to assist drivers at correcting the bad habits in their driving behavior, while offering helpful tips to improve fuel economy. In this work we take advantage of two key-points: the evolution of mobile terminals and the availability of a standard interface to access car data.)

- Kenneth L. Clarkson. 1985. Algorithms for Closest -Point Problems (Computational Geometry). Ph.D. Dissertation. Stanford University, Palo Alto, CA. UMI Order Number: AAT 8506171. (This dissertation reports a variety of new algorithms for solving closest point problems. The input to these algorithms is a set or sets of points in d-dimensional space, with an associated L (, p) metric.

- The problems considered are:(1) The all nearest neighbors problem. For point set A, find the nearest neighbors in A of each point in A.(2) The nearest foreign neighbor problem. For point sets A and B, find the closest point in B to each point in A. The geometric minimum spanning tree problem.) Goszczynska H., Kowalczyk L., Kuraszkiewicz B. (2014) Correlation Matrices as a ool to Analyze the Variability of EEG Maps.

- In: Piętka E., Kawa J., Wieclawek W. (eds) Information Technologies in Biomedicine, Volume 4. Advances in Intelligent Systems and Computing, vol 284. Springer. (The aim of this paper is to present the selected examples of possible applications of image of correlation coefficients matrix of EEG map series in the analysis of variation of the topography of the iso potential areas in EEG maps, and thus in the assessment of stationarity, spatio-temporal variability and trends of changes of bioelectric activity of the brain.

2.2 References

- Singh D, Singh M., "Internet of Vehicles for Smart and Safe Driving", International Conference on Connected Vehicles and Expo (ICCVE), Shenzhen, 19-23 Oct., 2015.

- Zhang, Y., Lin, W., and Chin, Y., "Data-Driven Driving Skill Characterization: Algorithm Comparison and Decision Fusion," SAE Technical Paper 2009 -01-1286, 2009, https://doi.org/10.4271/2009-01-1286.Azevedo, C. L Cardoso.

- J.E.Meseguer, C.T.Calafate, J.C.Canoand P.Manzoni,"Driving Styles: A smartphone application to assess driver behavior, "2013 IEEE Symposium on Computers and Communications (ISCC), Split, 2013, pp.000535- 000540.doi: 10.1109/ISCC.2013.6755001.

- Schneider, A,Hommel,G.,& Blettner,M.(2010). Linear Regression Analysis: Part 14 of a Series on Evaluation of Scientific Publications. Deutsches Ärzteblatt International, 107(44), pp. 776–782.

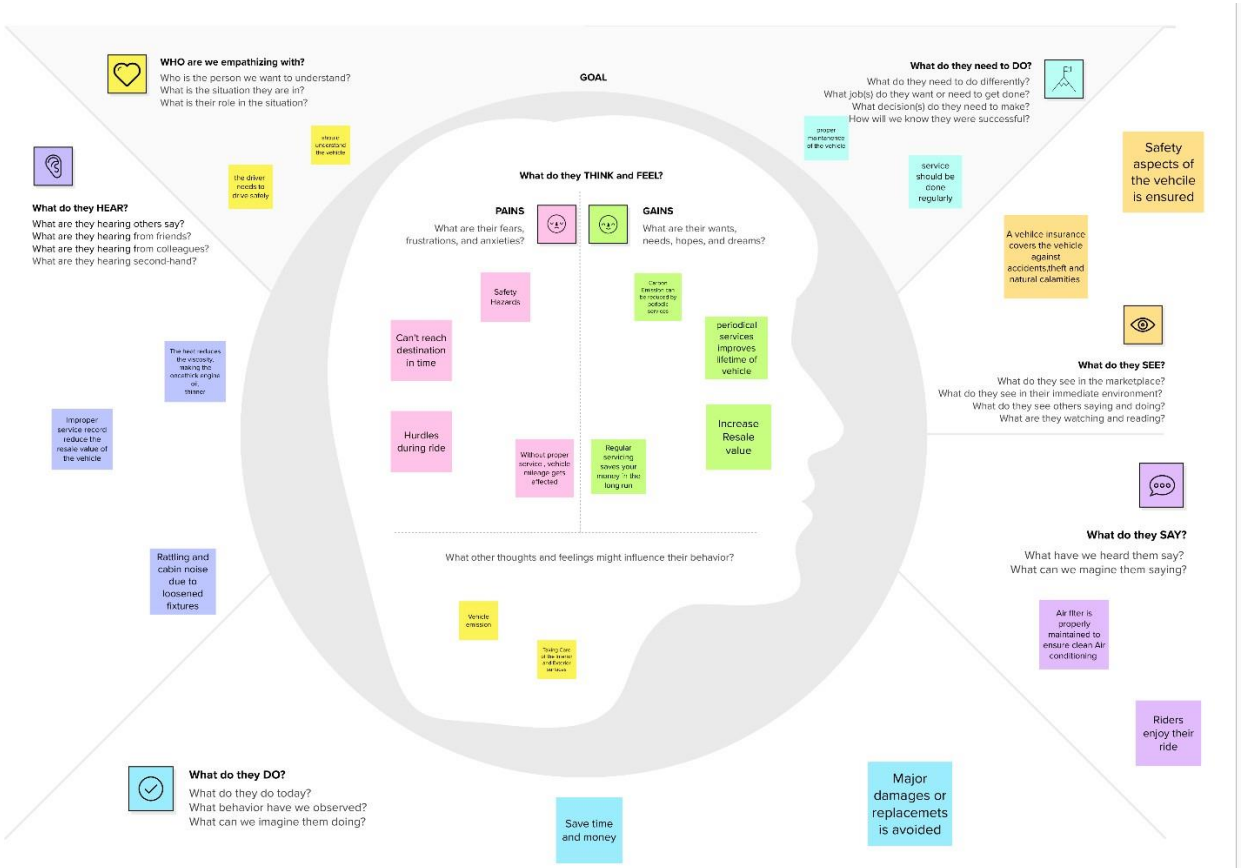
2.3 Problem Statement Definition

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Owner of a vehicle.	Collect the properties of the vehicle to predict the performance.	I don't know which properties to select.	I don't know which are the properties to be considered.	Confused.
PS-2	Owner of a vehicle.	Make predictions.	I don't know how the variables are correlated to each other.	There are many properties and what should I consider.	Little worried.

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

An empathy map is a collaborative tool teams can use to gain a deeper insight into their customers. Much like a user persona, an empathy map can represent a group of users, such as a customer segment.



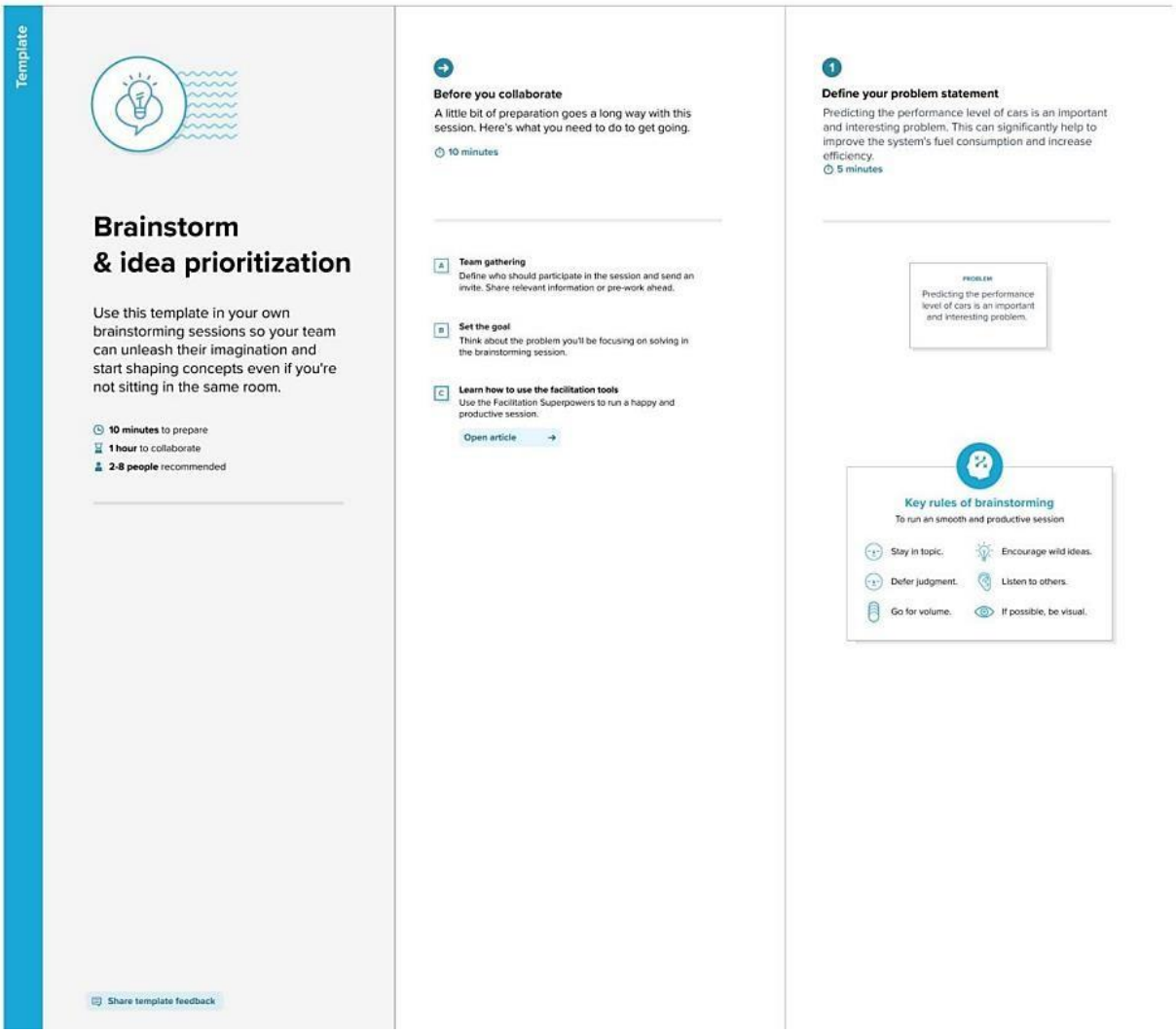
3.2 Ideation & Brainstorming

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem-solving.

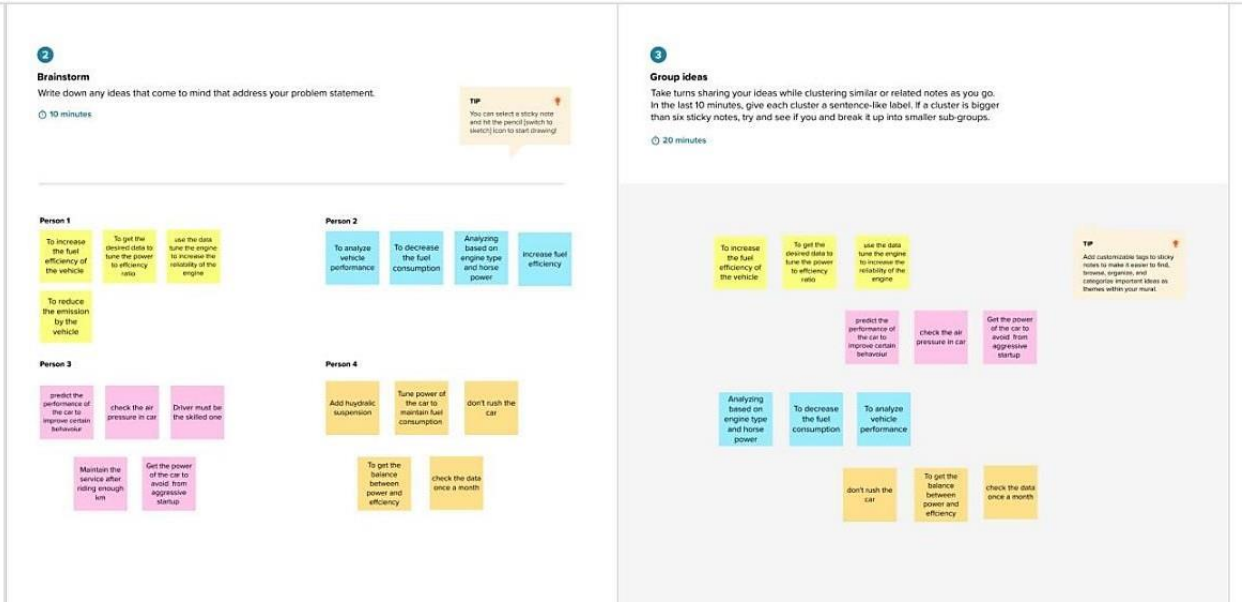
Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Use this template in your own brainstorming sessions so your team can unleash the imagination and start shaping concepts even if you're not sitting in the same room.

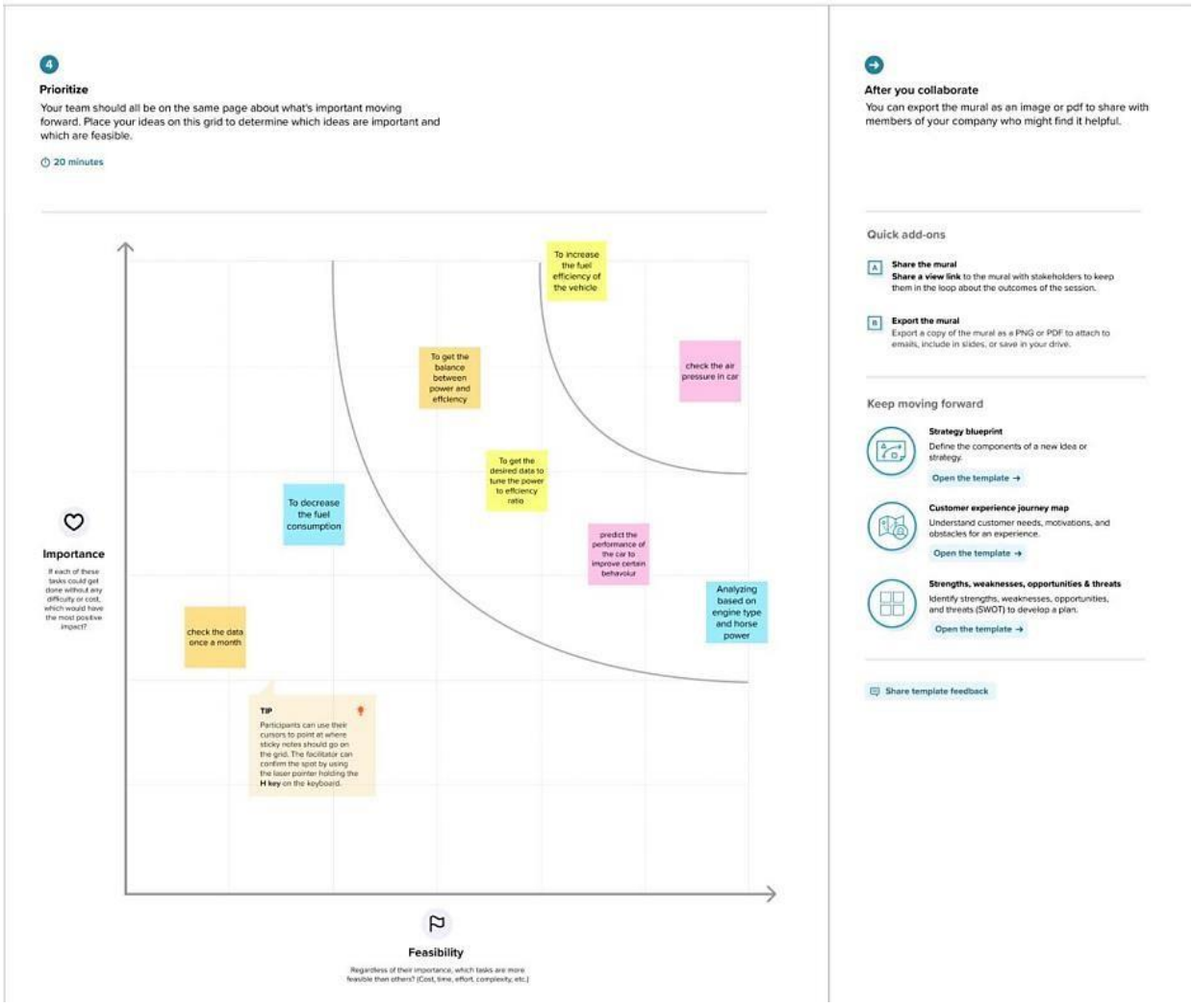
Step-1: Team Gathering, Collaboration and Select the Problem Statement:



Step-2: Brainstorm, Idea Listing and Grouping:



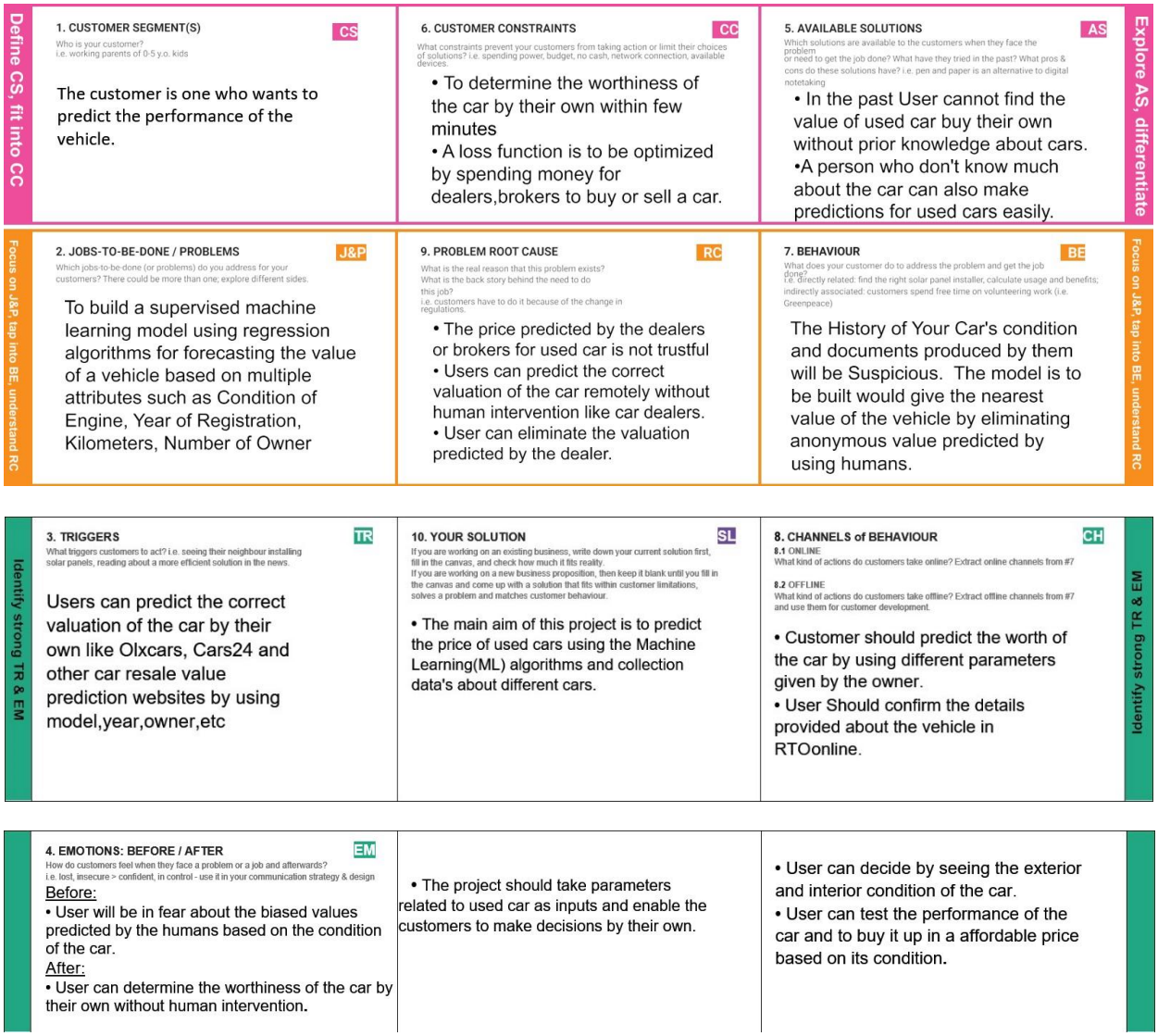
Step-3: Idea Prioritization:



3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	The objective of this project is to predict the price of used cars using the various Machine Learning models by using User Interface (UI).
2.	Idea / Solution description	To train the system with the dataset using a regression model and it will be integrated to the web-based application where the user is notified with the status.
3.	Novelty / Uniqueness	By using the optimal regression model to predict the value in a less amount to time and predict its value.
4.	Social Impact / Customer Satisfaction	The customer can get an idea about the resale value of their vehicle to predict the performance. By knowing the vehicle brand, fuel type, kilometeres driven .
5.	Business Model (Revenue Model)	The web-based application has a friendly UI for the customer to enter their vehicles detail and the system predicts the value within few seconds.
6.	Scalability of the Solution	Machine learning approaches, this project proposed a scalable framework for predicting values for different type of used cars. The solution given by the trained system is efficient and is nearly accurate value of the vehicle.

3.4 Problem Solution Fit:



4. REQUIREMENT ANALYSIS

4.1 Functional requirement:

Following are the functional requirements of the proposed solution:

FR No.	Functional Requirement (Epic)
FR-1	Enter the Inputs
FR-2	User Essential
FR-3	Data Prepossessing
FR-4	User input Evaluation
FR-5	Prediction

4.2 Non-Functional requirements:

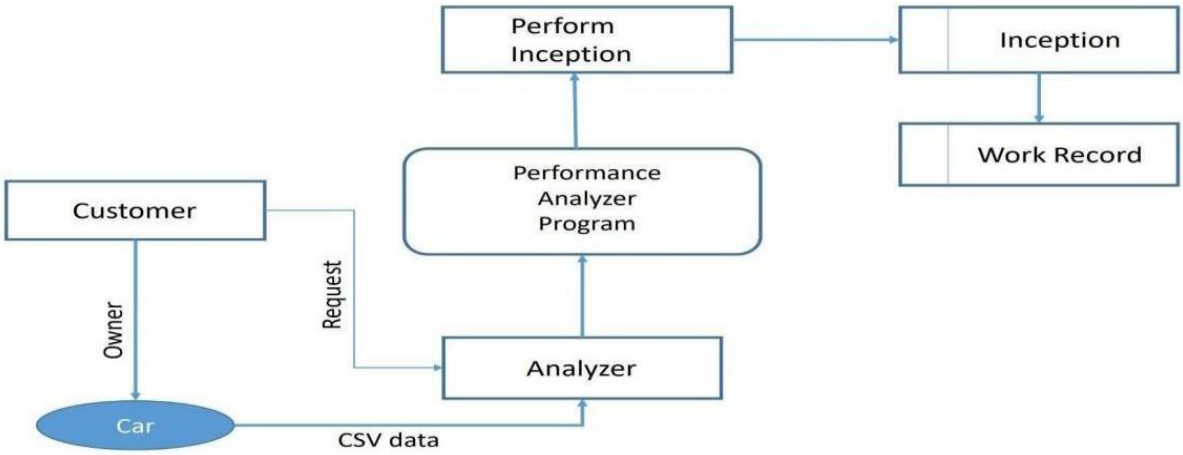
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	Get Inputs through a form	The analyzer allows the user to improve performance based on the results provided. It is easy to use with just the data required.
NFR-2	Predict the performance of the vehicle	The security is improved by using vehicle alarm, wheel lock, vehicle lock and also GPS tracker
NFR-3	Sample Dataset for training purpose	The reliability rating is good due to best performance, less frequency of problem occurrence and cost for repairing is low
NFR-4	Evaluating the given user values	The vehicle is upgraded in their quality and infrastructure to provide better performance like good mileage, smooth travel.
NFR-5	Fuel consumption and efficiency of the vehicle	The data required is collected by research persons and this data can be used to provide better results.

5. PROJECT DESIGN

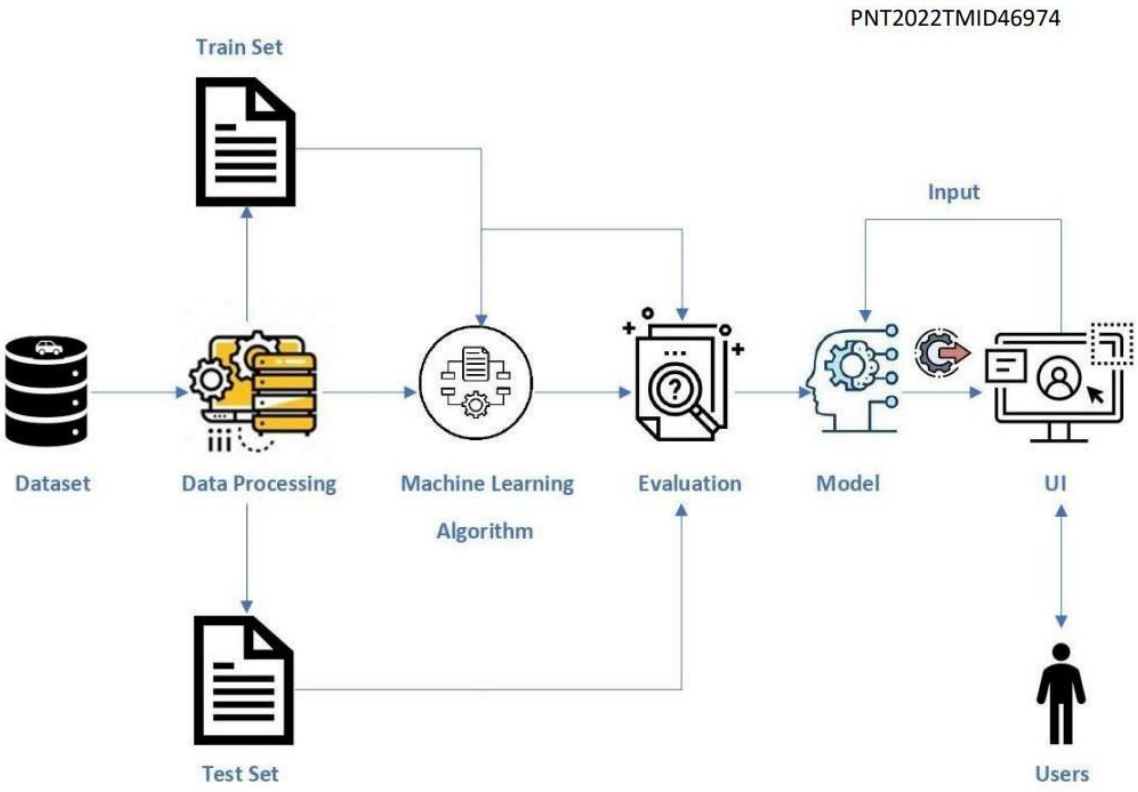
5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



5.2 Solution & Technical Architecture

A solution architecture (SA) is an architectural description of a specific solution. Solution Architectures combine guidance from different enterprise architecture viewpoints (business, information and technical), as well as from the enterprise solution architecture (ESA).



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Access the webpage	USN-1	As a user, anyone can access the webpage to check the specifications of the vehicle.	can access my webpage online at any time.	High	Sprint-1
Customer	Performance of the vehicle	USN-2	As per the usage of the user, the performance of the vehicle should be predicted easily.	Prediction can be done in an easy way.	High	Sprint-2
Customer	Accuracy to check the performance and health of the car	USN-3	By using our prediction, it helps to check the health ofthe car.	The efficiency of the car can be predicted	High	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Product Backlog, Sprint Schedule, and Estimation

Use the below template to create product backlog and sprint schedule

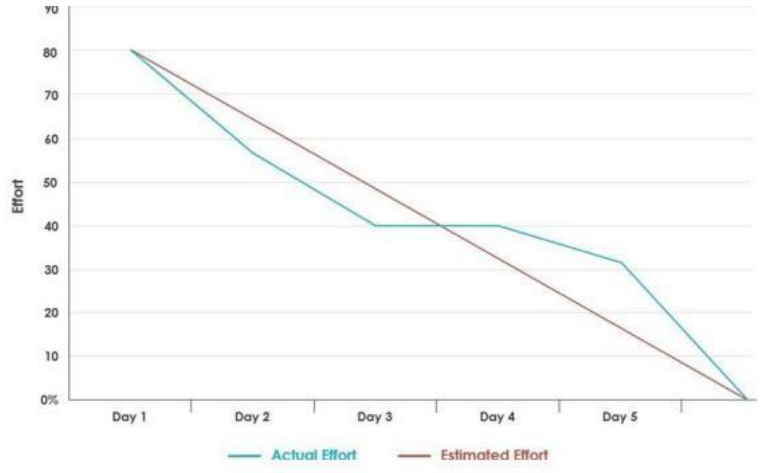
Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Data Preparation	USN-1	Collecting Car dataset and pre-processing it	10		
Sprint-1	Data Modeling	USN-2	create an ml model to predict the car Performance	5		
Sprint-1	Model Evaluation	USN-3	Calculate the performance, error rate, and complexity of ML model	5		
Sprint-2	Model Deployment	USN-4	Using flask and deploying model finally in IBM cloud using IBM storage and Watson Studio	20		
Sprint-3	Model prediction	USN-5	Predict the type of model. Here we have used random forest regression	10		
Sprint-3	Local deployment	USN-5	Local deployment will be done	5		

			while running the python fie in spyder			
Sprint-3	Application building	USN-6	As a user, I can give the necessary inputs	5		
Sprint-4	dashboard	USN-7	As a user, I can use the application by entering Car data	20		

Velocity:

Average Velocity = 80 / 20 = 4 Story Points per Day

BurnDown Chart:



7. CODING & SOLUTIONING

7.1 Feature 1

In [4]:
```
import numpy as np
import pandas as pd
```

In [5]:
```
#Reading The Dataset

datas = pd.read_csv(r"C:\Users\sunda\Desktop\IBM\car performance.csv")
datas.head()
```

Out[5]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino

In [6]:
```
# Handling Missing Values

datas.isnull().any()
```

Out[6]:
```
mpg             False
cylinders       False
displacement    False
horsepower      False
weight          False
acceleration    False
model year      False
origin          False
car name        False
dtype: bool
```

In [7]:
```
# Splitting The Dataset Into Dependent And Independent Variable.

x = datas.iloc[:,1:8].values
```

In [8]:
```
y = datas.iloc[:,0].values
```

In [9]:
```
# Split The Dataset Into Train Set And Test Set

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=0)
```

In [10]:
```
# Normalizing

from sklearn.preprocessing import StandardScaler
sd = StandardScaler()
x_train = sd.fit_transform(x_train)
x_test = sd.fit_transform(x_test)
```

In [11]:
```
x_train
```

Out[11]:
```
array([[ 1.49526939,  1.22961301,  1.24359144, ..., -0.79520768,
        -1.13752513, -0.73301171],
       [-0.85285735, -0.92367663, -1.16092059, ...,  1.24411524,
        -1.41177304,  0.5068698 ],
       [-0.85285735, -0.92367663, -0.68001818, ...,  0.05760009,
         1.05645814,  0.5068698 ],
       ...,
       [-0.85285735, -1.206235  , -1.45480539, ...,  1.42950823,
        -0.86327722,  0.5068698 ],
       [ 0.32120602,  0.56706235, -0.09224857, ..., -0.2390287 ,
        -1.41177304, -0.73301171],
       [-0.85285735, -0.99188037, -0.86703579, ...,  0.31715028,
        -0.31478141,  0.5068698 ]])
```

In [12]:
```
# Build The Model With The Random Forest Regressor

from sklearn.ensemble import RandomForestRegressor
d = RandomForestRegressor (n_estimators=30,random_state = 0)
d.fit(x_train,y_train)
```

Out[12]:
```
RandomForestRegressor(n_estimators=30, random_state=0)
```

In [13]:
```
# prediction

y_pred = d.predict(x_test)
y_pred
```

Out[13]:
```
array([14.38333333, 24.25666667, 14.21666667, 20.56666667, 18.47333333,
       30.21666667, 34.63333333, 21.15      , 16.30333333, 25.76      ,
       36.60333333, 36.27      , 19.53666667, 27.32333333, 16.54333333,
       32.99333333, 28.32333333, 27.49666667, 17.03      , 35.82      ,
       16.47333333, 23.54      , 23.16666667, 20.7       , 33.69666667,
       26.45      , 33.79666667, 30.37333333, 31.93666667, 16.57333333,
       20.26666667, 32.99      , 19.79666667, 34.08333333, 20.85666667,
       25.02      , 19.65333333, 17.14      , 34.78333333, 12.76666667,
       13.73333333, 15.2       , 28.32      , 32.76666667, 28.74333333,
       22.68666667, 20.54333333, 16.50666667, 23.38      , 29.88333333,
       34.31666667, 26.5       , 17.63      , 27.78333333, 15.96666667,
       12.96666667, 18.86666667, 26.91666667, 31.95666667, 15.68      ,
       20.81      , 25.97      , 19.84666667, 21.6       , 13.46666667,
       15.33333333, 14.2       , 18.90333333, 24.72666667, 14.21666667,
       34.87666667, 13.25      , 22.96666667, 18.77666667, 23.83333333,
       32.16666667, 28.17666667, 31.23666667, 31.94      , 14.35      ])
```

In [14]:
```
# Model Evaluation

from sklearn.metrics import r2_score
accuracy = r2_score(y_test,y_pred)
accuracy
```

Out[14]:
```
0.8914224071232417
```

7.1 Feature 2

In [54]:
```
import numpy as np
import pandas as pd
```

In [55]:
```
#Reading The Dataset
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='BqUs0KYfxIrLPRBCAnlsY1i-20vTS38c8Ks157IQ1BS7',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'vehicleperformanceanalyzer-donotdelete-pr-ccgjs2talld8s5'
object_key = 'car performance.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

datas = pd.read_csv(body)
datas.head()
```

Out[55]:

	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0	8	302.0	140	3449	10.5	70	1	ford torino

In [56]:
```
# Handling Missing Values

datas.isnull().any()
```

Out[56]:
```
mpg             False
cylinders       False
displacement    False
horsepower      False
weight          False
acceleration    False
model year      False
origin          False
car name        False
dtype: bool
```

In [57]:
```
# Splitting The Dataset Into Dependent And Independent Variable.

x = datas.iloc[:,1:8].values
```

In [58]:
```
y = datas.iloc[:,0].values
```

In [59]:
```
# Split The Dataset Into Train Set And Test Set

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size=0.2,random_state=0)
```

In [60]:
```
# Normalizing

from sklearn.preprocessing import StandardScaler
sd = StandardScaler()
x_train = sd.fit_transform(x_train)
x_test = sd.fit_transform(x_test)
```

In [61]:
```
x_train
```

Out[61]:
```
array([[ 1.49526939,  1.22961301,  1.24359144, ..., -0.79520768,
        -1.13752513, -0.73301171],
       [-0.85285735, -0.92367663, -1.16092059, ...,  1.24411524,
        -1.41177304,  0.5068698 ],
       [-0.85285735, -0.92367663, -0.68001818, ...,  0.05760009,
         1.05645814,  0.5068698 ],
       ...,
       [-0.85285735, -1.206235  , -1.45480539, ...,  1.42950823,
        -0.86327722,  0.5068698 ],
       [ 0.32120602,  0.56706235, -0.09224857, ..., -0.2390287 ,
        -1.41177304, -0.73301171],
       [-0.85285735, -0.99188037, -0.86703579, ...,  0.31715028,
        -0.31478141,  0.5068698 ]])
```

In [62]:
```
# Build The Model With The Random Forest Regressor

from sklearn.ensemble import RandomForestRegressor
d = RandomForestRegressor (n_estimators=30,random_state = 0)
d.fit(x_train,y_train)
```

Out[62]:
```
RandomForestRegressor(n_estimators=30, random_state=0)
```

In [63]:
```
# prediction

y_pred = d.predict(x_test)
y_pred
```

Out[63]:
```
array([14.38333333, 24.25666667, 14.21666667, 20.56666667, 18.47333333,
       30.21666667, 34.63333333, 21.15      , 16.30333333, 25.76      ,
       36.60333333, 36.27      , 19.53666667, 27.32333333, 16.54333333,
       32.99333333, 28.32333333, 27.49666667, 17.03      , 35.82      ,
       16.47333333, 23.54      , 23.16666667, 20.7       , 33.69666667,
       26.45      , 33.79666667, 30.37333333, 31.93666667, 16.57333333,
       20.26666667, 32.99      , 19.79666667, 34.08333333, 20.85666667,
       25.02      , 19.65333333, 17.14      , 34.78333333, 12.76666667,
       13.73333333, 15.2       , 28.32      , 32.76666667, 28.74333333,
       22.68666667, 20.54333333, 16.50666667, 23.38      , 29.88333333,
       34.31666667, 26.5       , 17.63      , 27.78333333, 15.96666667,
       12.96666667, 18.86666667, 26.91666667, 31.95666667, 15.68      ,
       20.81      , 25.97      , 19.84666667, 21.6       , 13.46666667,
       15.33333333, 14.2       , 18.90333333, 24.72666667, 14.21666667,
       34.87666667, 13.25      , 22.96666667, 18.77666667, 23.83333333,
       32.16666667, 28.17666667, 31.23666667, 31.94      , 14.35      ])
```

In [64]:
```
!pip install ibm_watson_machine_learning
```

```
Requirement already satisfied: ibm_watson_machine_learning in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (1.0.257)
Requirement already satisfied: importlib-metadata in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (4.8.2)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2.26.0)
Requirement already satisfied: pandas<1.5.0,>=0.24.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (1.3.4)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (0.8.9)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (1.26.7)
Requirement already satisfied: ibm-cos-sdk==2.11.* in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2.11.0)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (2022.9.24)
Requirement already satisfied: packaging in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (21.3)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm_watson_machine_learning) (0.3.3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (0.10.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (2.11.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (2.8.2)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm_watson_machine_learning) (2021.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas<1.5.0,>=0.24.2->ibm_watson_machine_learning) (1.20.3)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->ibm-cos-sdk-core==2.11.0->ibm-cos-sdk==2.11.*->ibm_watson_machine_learning) (1.15.0)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->ibm_watson_machine_learning) (3.3)


Requirement already satisfied: zipp>=0.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from importlib-metadata->ibm_watson_machine_learning) (3.6.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from packaging->ibm_watson_machine_learning) (3.0.4)
```

In [65]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url": "https://us-south.ml.cloud.ibm.com",
                   "apikey": "APGWCp5rv2AMp9x_hadO6NK6pzJcZAOadcl0zoTjFHyG"
                  }
client = APIClient(wml_credentials)
```

In [66]:
```
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    #print(space)
    return(next(item for item in space['resources'] if item['entity']["name"] == space_name )['metadata']['id'])
```

In [67]:
```
space_uid = guid_from_space_name(client, 'models')
print("Space UID = " + space_uid)
```

```
Space UID = 8e9ce932-c021-4332-a82c-9994699d398a
```

In [68]:
```
client.set.default_space(space_uid)
```

Out[68]:
```
'SUCCESS'
```

8.Testing

8.1 Test Cases

GIF Gesellschaft für Industry for schung is an automotive company based out of Germany, covering the vehicle transmissions and the design, testing, and development of powertrain system. GSA system, which was developed by GIF, has proven in many years of testing to be a valuable tool, mainly by complex transmission and gear development projects. GSA system is not only applied for objectively assessing the change of gears in a vehicle, but also permits measuring and analyzing the influencing factors of gear shifting quality.

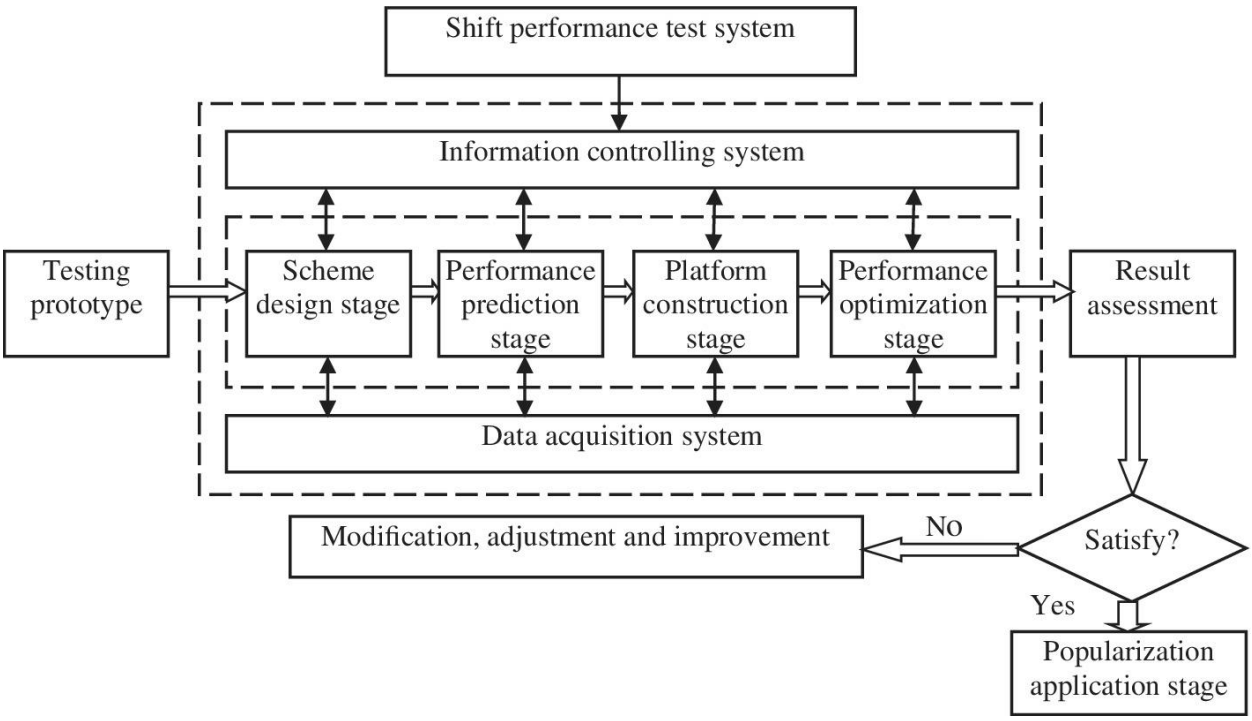
Therefore, during the new development and integration of transmissions, transmission component testing, and quality assurance, GSA system is indispensable as the right tool. Based on GSA shifting performance test evaluation system, this project completes the control strategy research of improving the shifting quality of automobile transmission. The technical route to be adopted. The development process of transmission shifting performance is mainly divided into scheme design stage, performance prediction stage, platform construction stage, performance optimization stage, and popularization application stage.

First, the research idea is clear, the research scheme is determined, and the theoretical research framework of shifting test and evaluation is constructed. Second, relevant materials are collected, the performance parameters of transmission and shifting control mechanism are summarized, and the shifting performance of the whole system is predicted. Third, the shifting control performance bench test and GSA test system are built to carry out the subjective and objective test and research of shifting performance, and the real-time acquisition of shifting performance is carried out.

The performance indicators in the process are analyzed and evaluated to provide data support for improving the shifting performance of the system. Then the optimization improvement measures are specified and the shifting performance is re-evaluated to obtain the best matching relationship of the shifting performance and verify the effectiveness of the 25 control strategy research scheme. Finally, the shifting optimization test and analysis path based on GSA test technology is formed.

GSA test analysis means that the whole test system includes hardware acquisition equipment and software processing analysis tools. Through real-time measurement and analysis of the force, travel, acceleration, and other important parameters of the

shifting lever knob, the performance of the whole vehicle is systematically evaluated from the aspects of the shifting and selection force (travel), system stiffness, free play, and dynamic impact.



8.2 User Acceptance Testing (UAT)

It is a process to check that system accepts requirements of a user or not. It's performed at a time when the system used by actual users. User acceptance testing comes after →Unit Testing →Integration Testing →System Testing →Acceptance Testing in the process of testing.

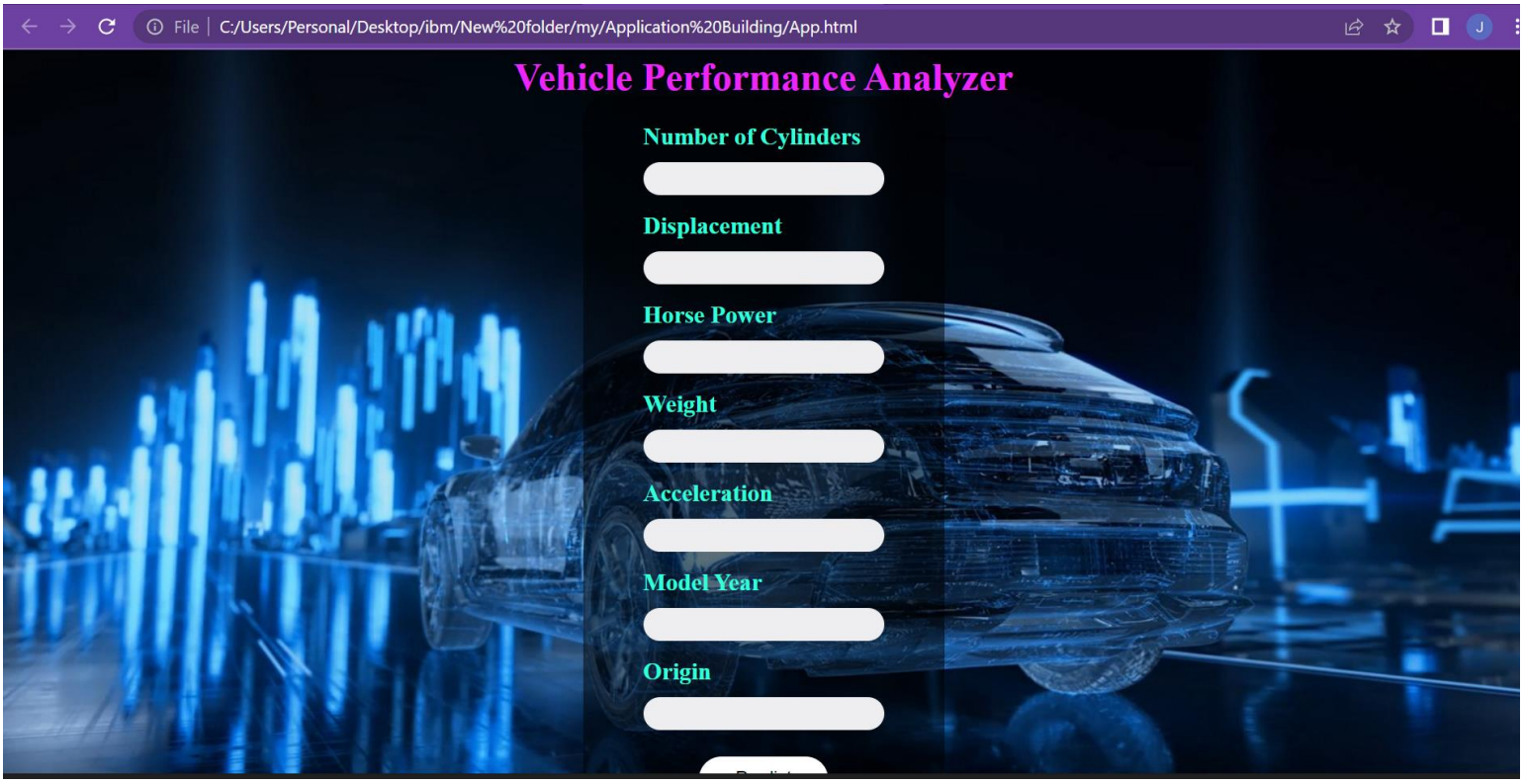
Alpha Testing

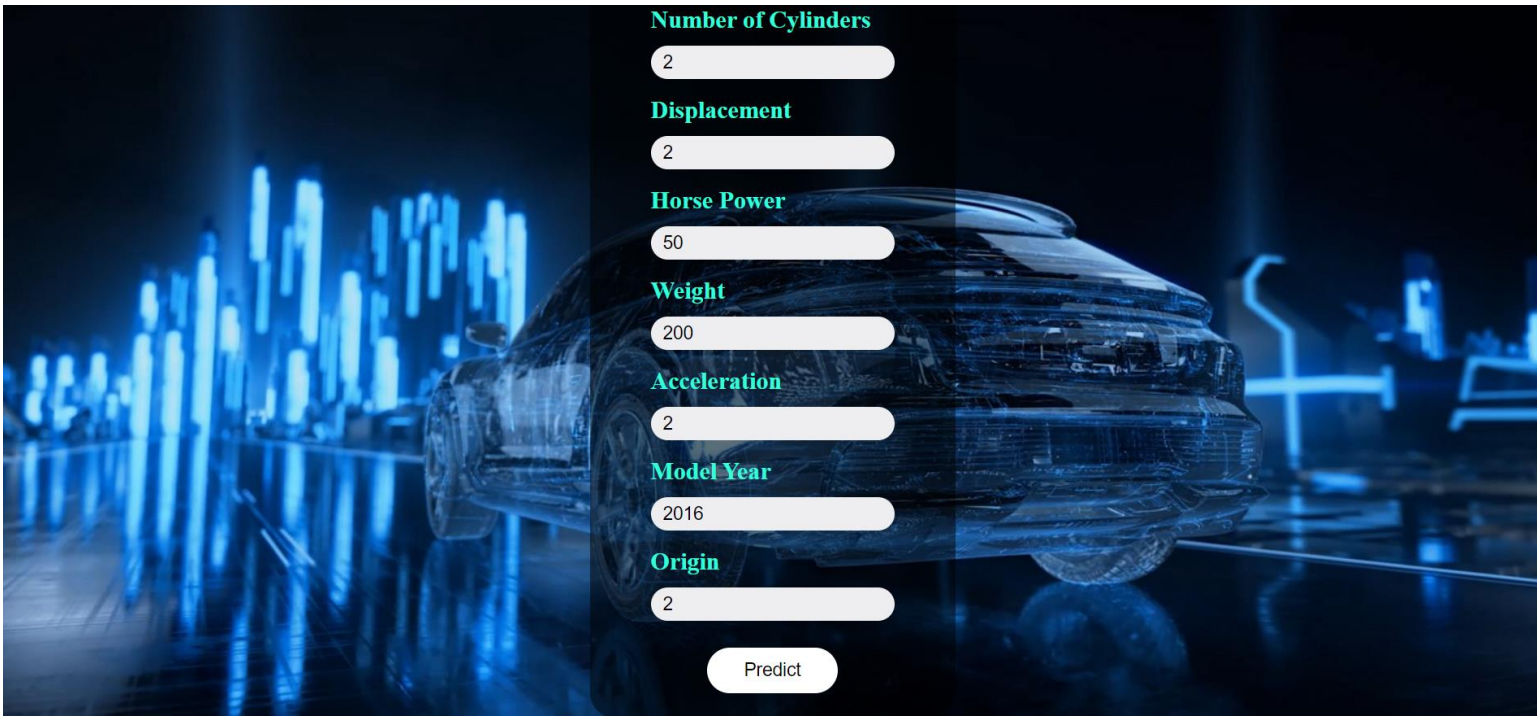
It is the type of testing, executed to identify all possible issues/bugs before releasing a product to every data users or public. It simulates real users by using Black Box and White Box testing techniques. The primary function is to carry out tasks executed by a typical user. Alpha testing carried out in a developed environment or in a lab environment where a product developed and the user of a product are internal employees of an organization.

Beta Testing

It is the type of testing in which users of software or application are real users. In this application is tested in a real environment and considered as a form of an external User Acceptance testing. Beta version of a software released to a limited number of a user when tested in a Real-Time environment with the help of real users, to obtain feedback on product quality. Beta testing reduces failures, risks and provides increased quality of a product through customer validation. It is a final test before shipping a product to the customers. In this type of testing, getting direct feedback from users is a significant benefit. It is required to test a product finally in a Real-Time environment.

9. RESULTS


File | C:/Users/Personal/Desktop/ibm/New%20folder/my/Application%20Building/App.html
Vehicle Performance Analyzer
Number of Cylinders
Displacement
Horse Power
Weight
Acceleration
Model Year
Origin


Number of Cylinders
2
Displacement
2
Horse Power
50
Weight
200
Acceleration
2
Model Year
2016
Origin
2
Predict

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Easy Implementation
- Low cost
- Can know and maintain the driver and car performance

DISADVANTAGE

- It is very difficult to find the place for placing temperature sensor and pressure sensor.

11. CONCLUSION

In this paper we have obtained some newer insights about the car data analysis such as economic driving index (ECN_DRVG_INDX) and safety driving index (SFTY_DRVG_INDX.) The results have proven to be approximately 80% fitting the given features and are very helpful to be used in different use cases such as a parameter in finding the driver's driving performance in a driving school, as a good estimate for finding an optimal price for a used car that can be based on several factors which we have analyzed in this paper etc. We also found that the model used to train the data can be improved further by finding better hyper parameter values for the features. It is also possible that different features can be considered for improving the hypothesis**.**

12. FUTURE SCOPE

There is more scope in future for research and analysis of fuel efficiency by including other factors like the road condition and real-time traffic with the help of google maps, this would help in analyzing much deeper. The knowledge discovered from the research and future work can be used by the car manufacturing companies to improve the fuel economy by considering the characteristics that substantially influence the fuel efficiency.

13. APPENDIX

GitHub link: https://github.com/IBM-EPBL/IBM-project-41248-1660640630