GANADIPATHY TULSI'S JAIN ENGINEERING COLLEGE

kaniyambadi, Vellore -632102

IBM-PROJECT-54094-166159431

MACHINE LEARNING BASED VEHICLE PERFORMANCE ANALYZER

PROJECT REPORT

Submitted by

PNT2022TMID39717

Team Leader	**-**	**S. DEEPAK**
Team Member 1	**-**	**B. KAMALESH**
Team Member 2	**-**	**R .PRIYANKHA**
Team Member 3	**-**	**P.SUDHARSHAN**

Project Mentor	Project Evaluator	Project spoc
Mrs.S.Priya	Mrs.G.Sundari	Mr.D.DuraiKumar

Industry Mentor Ms .NIDHI

1.INTRODUCTION

1.1 Project Overview:

Predicting the performance level of cars is an important and interesting problem. The main goal is to predict the performance of the car to improve certain behaviours of the vehicle. This can significantly help to improve the system's fuel consumption and increase efficiency. The performance analysis of the car is based on the engine type, no of engine cylinders, fuel type, horsepower, etc. These are the factors on which the health of the car can be predicted. It is an ongoing process of obtaining, researching, analyzing, and recording health based on the above three factors. The performance objectives like mileage, dependability, flexibility and cost can be grouped together to play a vital role in the prediction engine and engine management system. This approach is a very important step towards understanding the vehicle's performance. so this project aims at building a Machine Learning [ML] model to predict vechile perfomace analyzer.

1.2 Purpose:

The main goal is to predict the performance of the car to improve the certain behavior of the vehicle. This can significantly help to improve the system's fuel consumption and increase efficiency. The performance analysis of the car is based on the engine type, no of engine cylinders, fuel type, and horsepower, etc.

2.LITERATURE SURVEY

2.1 Existing Problem:

In the existing system, implementation of machine learning algorithm is bit complex to build due to the lack of information about the data visualization. Mathematical calculations are used in exisiting system for model building this may takes the lot of time and complexity. To overcome this all, we use machine learning packages available in the scikit-learn library.

Disadvantage:

- High complexity
- Time complexity

2.2 References:

1. 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.
2. Zhang, Y., Lin, W., and Chin, Y., "Data-Driven Driving Skill Characterization: Algorithm Comparison and Decision Fusion," SAE Technical Paper 2009-01-1286, 2009, C. L Cardoso.
3. 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.
4. 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.

5. Kenneth L. Clarkson. 1985. *Algorithms for Closest-Point Problems (Computational Geometry).* Ph.D. Dissertation. Stanford University, Palo Alto, CA. UMI Order Number: AAT 8506171.

6. 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.

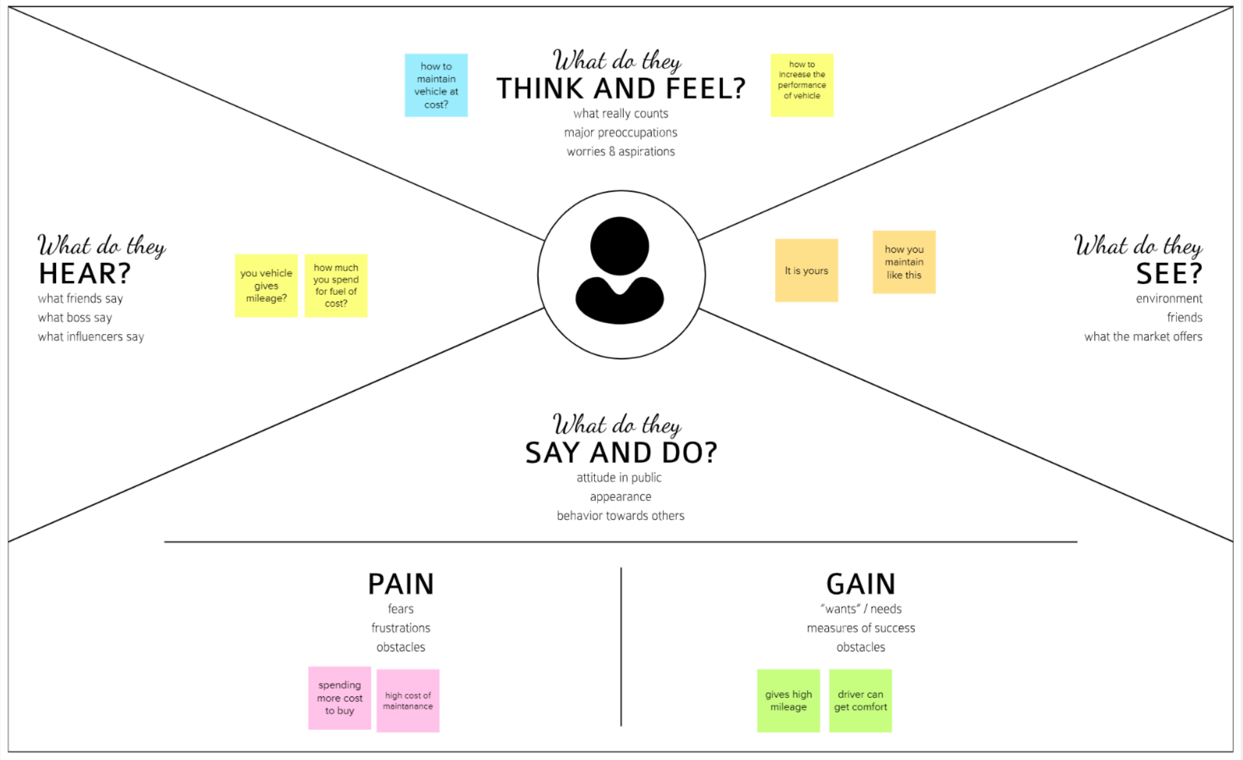
7. Goszczynska H., Kowalczyk L., Kuraszkiewicz B. (2014) Correlation Matrices as a Tool to Analyze the Variability of EEG Maps. In: Piętka E., Kawa J., Wieclawek W. (eds) *Information Technologies in Biomedicine*, Volume 4. Advances in Intelligent Systems and Computing, vol 284. Springer.

2.3 Problem Statement Definiton:

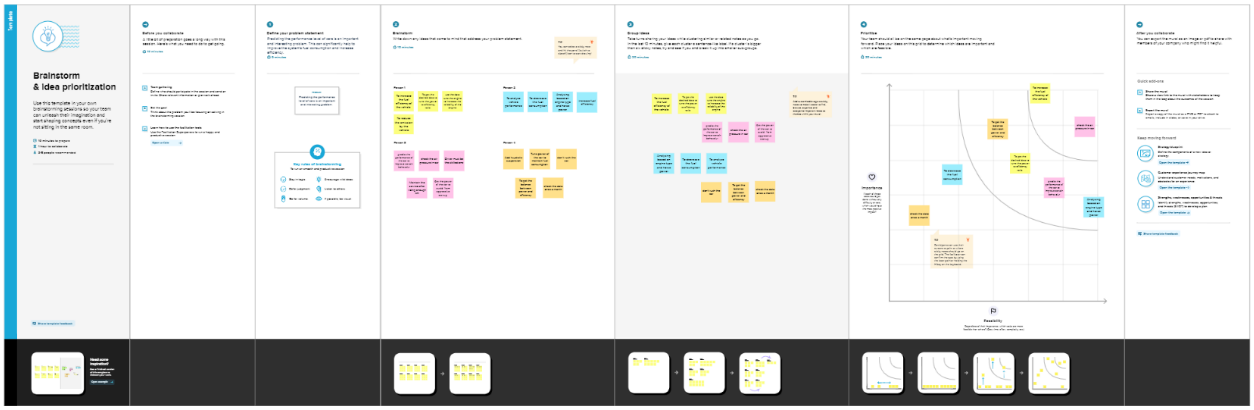
Vehicle performance is the study of the motion of a vehicle The motion of any vehicle depends upon all the forces and moments that act upon it These forces and moments, for the most part are caused by interaction of the vehicle with the surrounding medium(s)such as air or water (eg fluid static and dynamic forces), gravitational, Earth’s surface (support, ground, or landing gear forces), and on-board energy consuming devices such as rocket, turbojet, piston engine and propellers Consequently, in order to fully understand the performance problem, it is necessary to study and, in some way, characterize these interacting forces Generally speaking, the performance of a vehicle can be evaluated using following indicators: the maximal speed that can be reached, the accelerating time from zero to a certain speed, milege,weight,etc.To solve this problem we will develop the models using differnt algorithms thst produce more accuracy.

3. IDEATION & PROPOSED SOLUTION

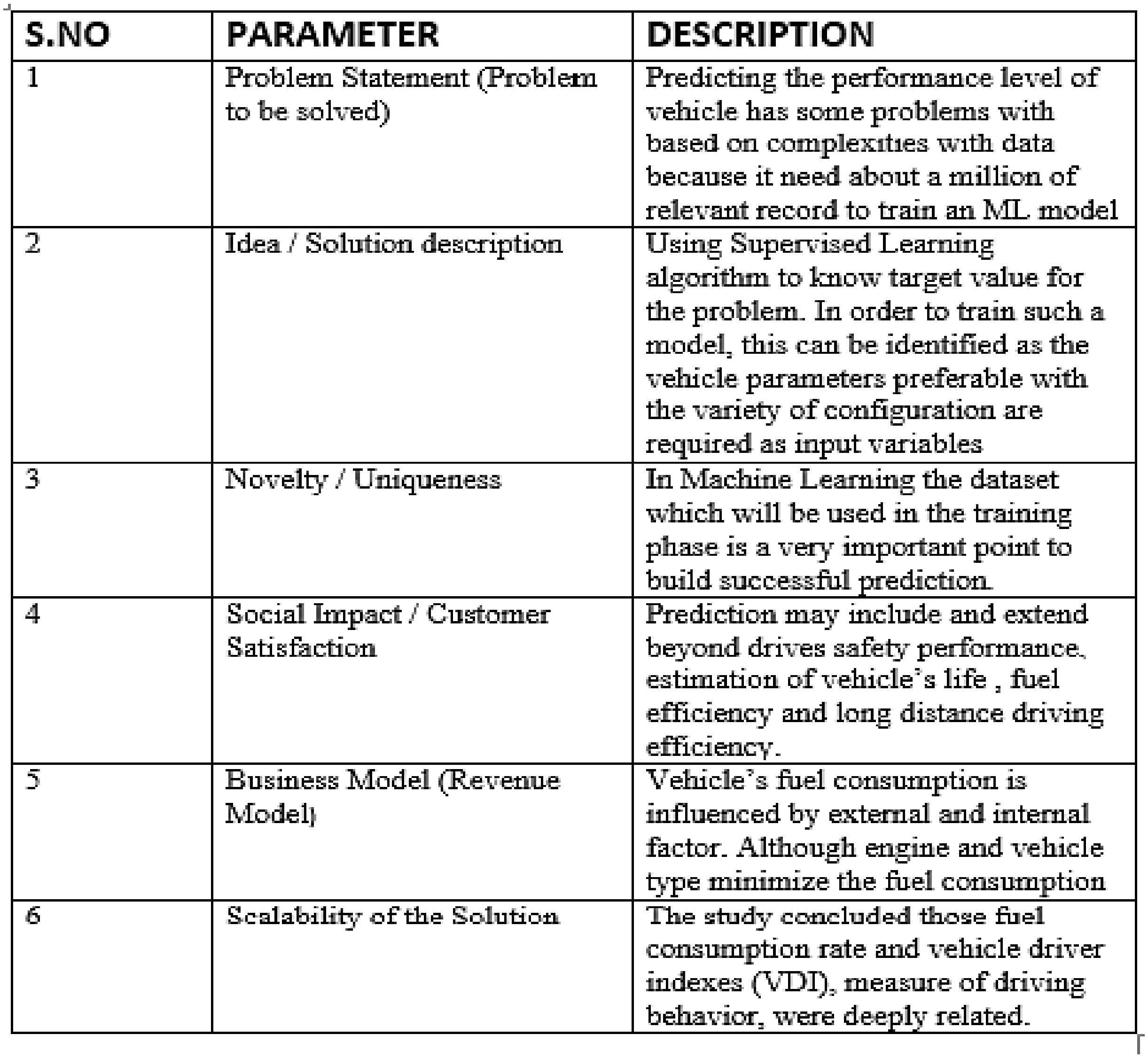
3.1 Empathy Map Canvas:



3.2 Ideation & Brainstorming:

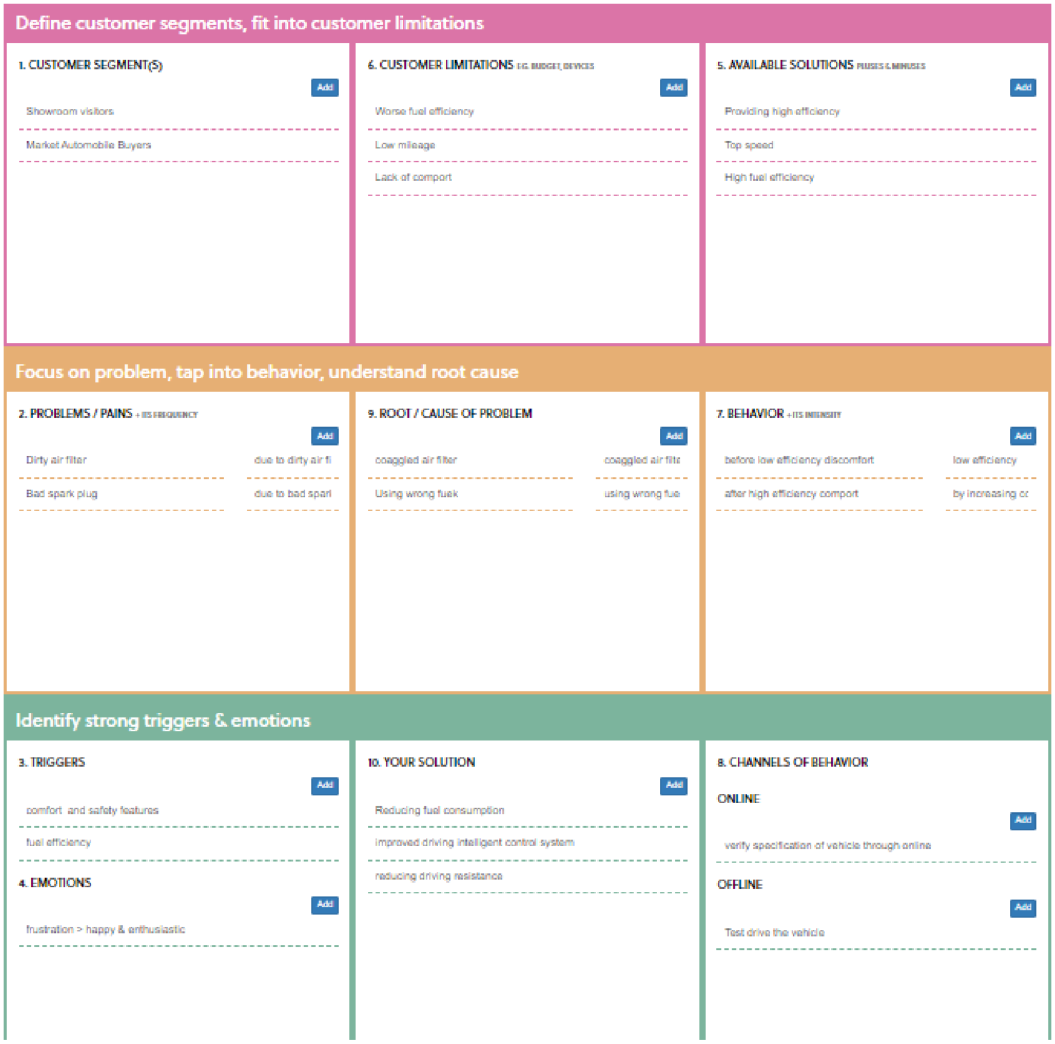

Template
Brainstorm
& Idea prioritization
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.
Before you collaborate
Define your problem statement
Brainstorm
Group ideas
Prioritize
After you collaborate

3.3 Proposed Solution:



S.NO	PARAMETER	DESCRIPTION
1	Problem Statement (Problem to be solved)	Predicting the performance level of vehicle has some problems with based on complexities with data because it need about a million of relevant record to train an ML model
2	Idea / Solution description	Using Supervised Learning algorithm to know target value for the problem. In order to train such a model, this can be identified as the vehicle parameters preferable with the variety of configuration are required as input variables
3	Novelty / Uniqueness	In Machine Learning the dataset which will be used in the training phase is a very important point to build successful prediction.
4	Social Impact / Customer Satisfaction	Prediction may include and extend beyond drives safety performance, estimation of vehicle's life , fuel efficiency and long distance driving efficiency.
5	Business Model (Revenue Model)	Vehicle's fuel consumption is influenced by external and internal factor. Although engine and vehicle type minimize the fuel consumption
6	Scalability of the Solution	The study concluded those fuel consumption rate and vehicle driver indexes (VDI), measure of driving behavior, were deeply related.

3.4 Problem Solution Fit



Define customer segments, fit into customer limitations

1. CUSTOMER SEGMENT(S)

- Showroom visitors
- Market Automobile Buyers

6. CUSTOMER LIMITATIONS E.G. BUDGET, DEVICES

- Worse fuel efficiency
- Low mileage
- Lack of comport

5. AVAILABLE SOLUTIONS PLUSES & MINUSES

- Providing high efficiency
- Top speed
- High fuel efficiency

Focus on problem, tap into behavior, understand root cause

2. PROBLEMS / PAINS + ITS FREQUENCY

- Dirty air filter — due to dirty air fi
- Bad spark plug — due to bad spari

9. ROOT / CAUSE OF PROBLEM

- coaggled air filter — coaggled air filte
- Using wrong fuek — using wrong fue

7. BEHAVIOR + ITS INTENSITY

- before low efficiency discomfort — low efficiency
- after high efficiency comport — by increasing cc

Identify strong triggers & emotions

3. TRIGGERS

- comfort and safety features
- fuel efficiency

4. EMOTIONS

- frustration > happy & enthusiastic

10. YOUR SOLUTION

- Reducing fuel consumption
- improved driving intelligent control system
- reducing driving resistance

8. CHANNELS OF BEHAVIOR

ONLINE

- verify specification of vehicle through online

OFFLINE

- Test drive the vehicle

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS:

FR.NO	**FUNCTIONAL REQUIREMENTS**	**DESCRIPTION**
FR-1	Enter the inputs	Get the inputs through form
FR-2	User Essentials	Predict the performance of the car
FR-3	Data Preprocessing	Sample dataset for training purpose
FR-4	User input evaluation	Evaluating the given user values
FR-5	Prediction	Fuel consumption and efficiency of the vehicle

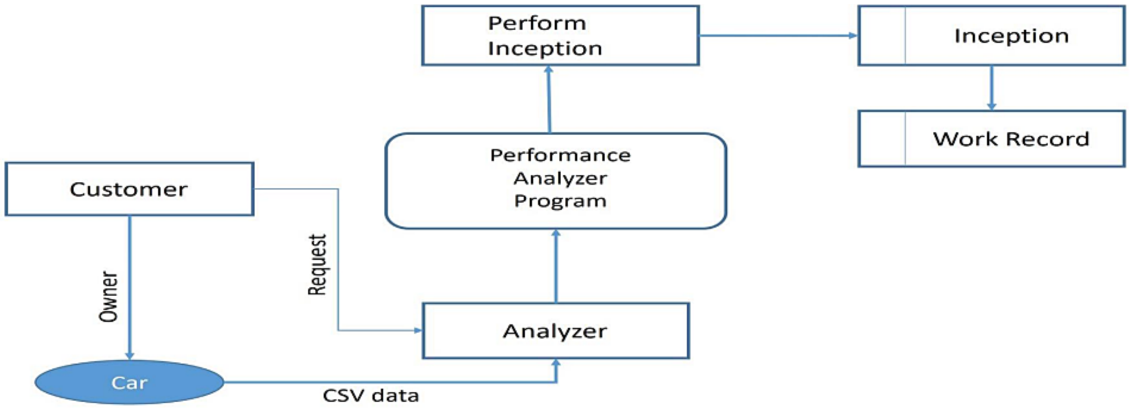
4.2 NON-FUNCTIONAL REQUIREMENTS:

FR.NO	**NON-FUNCTIONAL REQUIREMENT**	**DESCRIPTION**
NFR-1	Usability	The customers can have the opportunity to view a better interpretation of results. the customer are also recommended with prediction techniques based on performance
NFR-2	Security	We have designed this project to secure the people from buying low performance vehicle
NFR-3	Reliability	The reliability should be in providing the efficient and high performance of the vehicle

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:

Solution:

predicting the performance level of cars is an important and interesting problem. The main goal of the current study is to predict the performance of the car to improve certain behavior of the vehicle. This can significantly help to improve the systems fuel consumption and increase the efficiency. The performance analysis of the car based on theengine type, no of engine cylinders, fuel type and horsepower etc.These are the factors on which the health of the car can be predicted. It is an on-going process of obtaining, researching, analyzing and recording thehealth based on the above three factors. The performance objectives likemileage, dependability, flexibility and cost can be grouped together to play a vital role in prediction engine and engine management system.
This approach is the very important step towards understanding thevehicles performance

It is an important to analyse the factors using number of well-known approaches of machine learning algorithms like linear regression, decision tree and random forest to improve the vehicle performance efficiency.The range, durability and longevity of automotive traction batteries are ‘hot topics’ in automotive engineering. And here we consider a performance in mileage. To solve this problem, we will develop the models, using the different algorithms and neural networks. We will then see which algorithm predicts car performance(Mileage) withhigher accuracy.

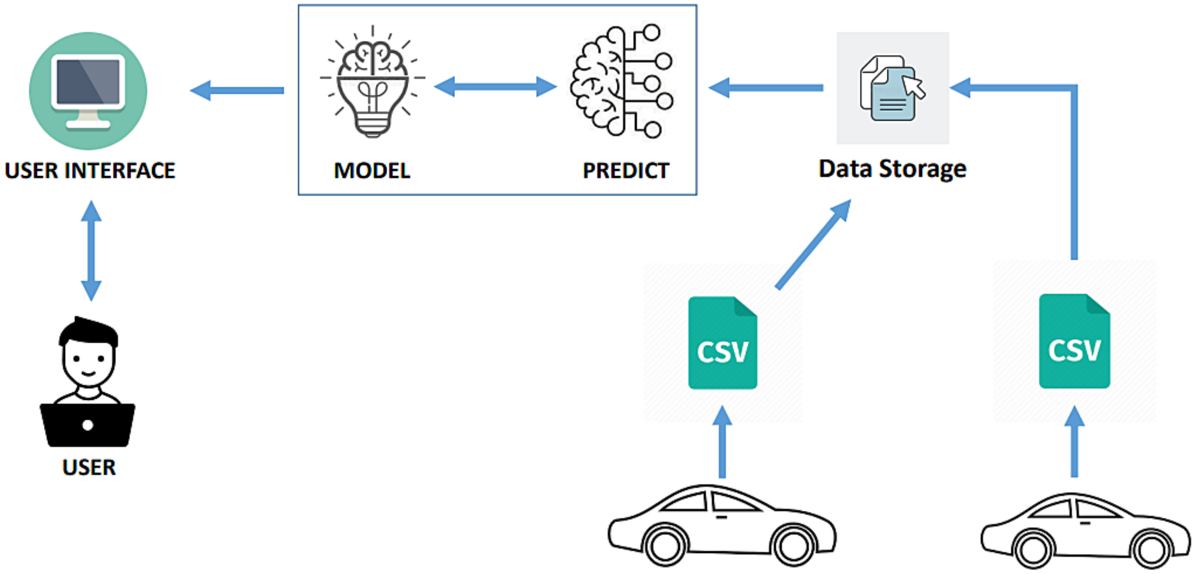


fig.5.2.1 solution architecture

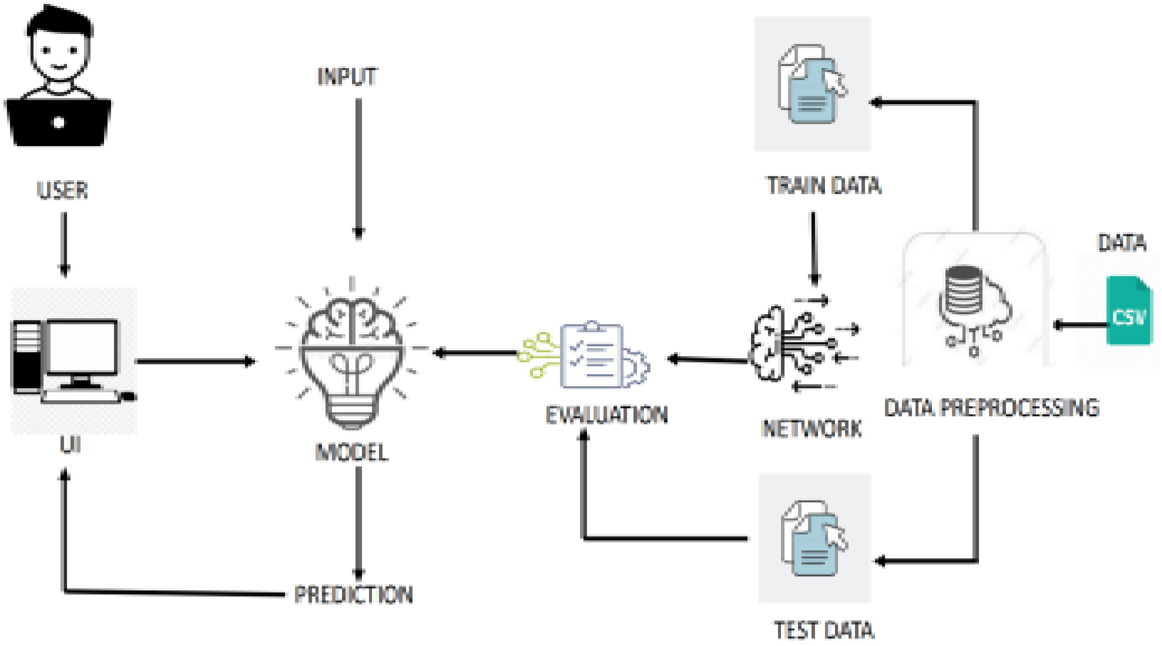
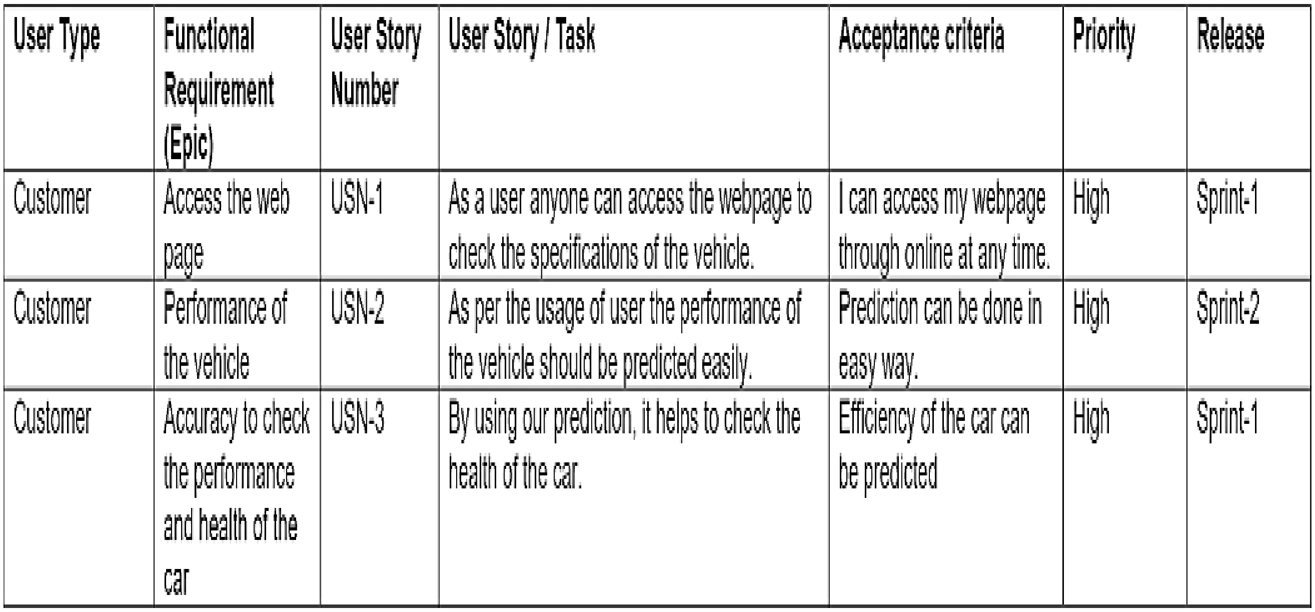


fig 5.2.2 Technical Architecture

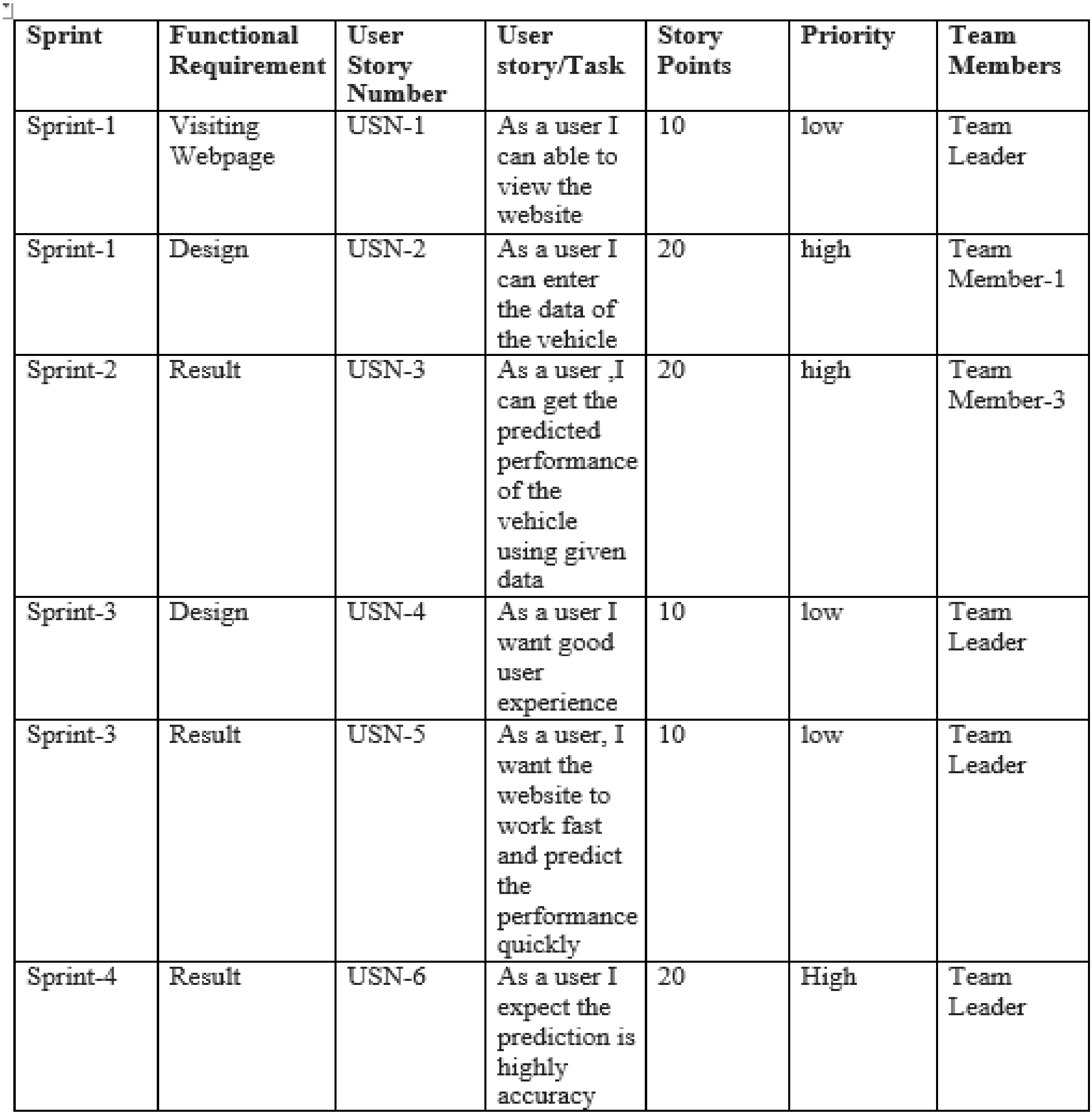
5.3 User Stories:



User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Access the web page	USN-1	As a user anyone can access the webpage to check the specifications of the vehicle.	I can access my webpage through online at any time.	High	Sprint-1
Customer	Performance of the vehicle	USN-2	As per the usage of user the performance of the vehicle should be predicted easily.	Prediction can be done in 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 of the car.	Efficiency of the car can be predicted	High	Sprint-1

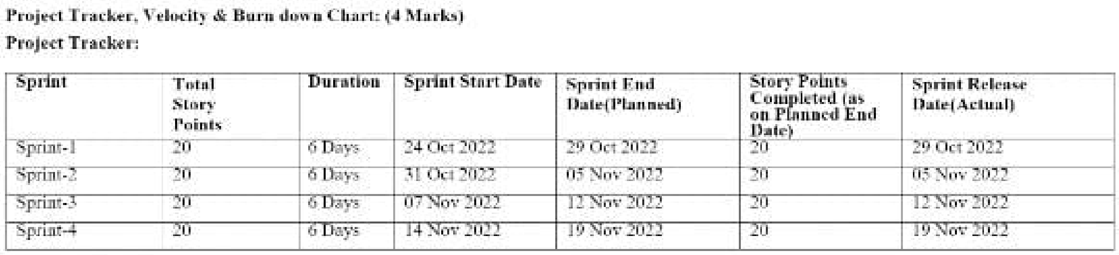
6. Project Planning & Scheduling

6.1Sprint Planningand Estimation



Sprint	Functional Requirement	User Story Number	User story/Task	Story Points	Priority	Team Members
Sprint-1	Visiting Webpage	USN-1	As a user I can able to view the website	10	low	Team Leader
Sprint-1	Design	USN-2	As a user I can enter the data of the vehicle	20	high	Team Member-1
Sprint-2	Result	USN-3	As a user ,I can get the predicted performance of the vehicle using given data	20	high	Team Member-3
Sprint-3	Design	USN-4	As a user I want good user experience	10	low	Team Leader
Sprint-3	Result	USN-5	As a user, I want the website to work fast and predict the performance quickly	10	low	Team Leader
Sprint-4	Result	USN-6	As a user I expect the prediction is highly accuracy	20	High	Team Leader

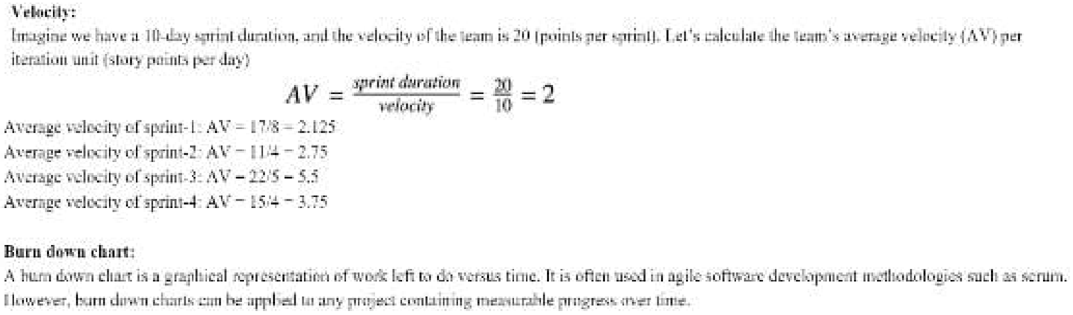
6.2 Sprint Delivery Schedule



Project Tracker, Velocity & Burn down Chart: (4 Marks)

Project Tracker:

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date(Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022



Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{\textit{sprint duration}}{\textit{velocity}} = \frac{20}{10} = 2$$

Average velocity of sprint-1: AV = 17/8 = 2.125

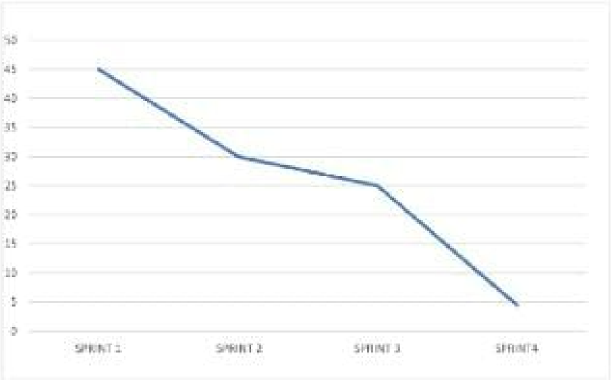
Average velocity of sprint-2: AV = 11/4 = 2.75

Average velocity of sprint-3: AV = 22/5 = 5.5

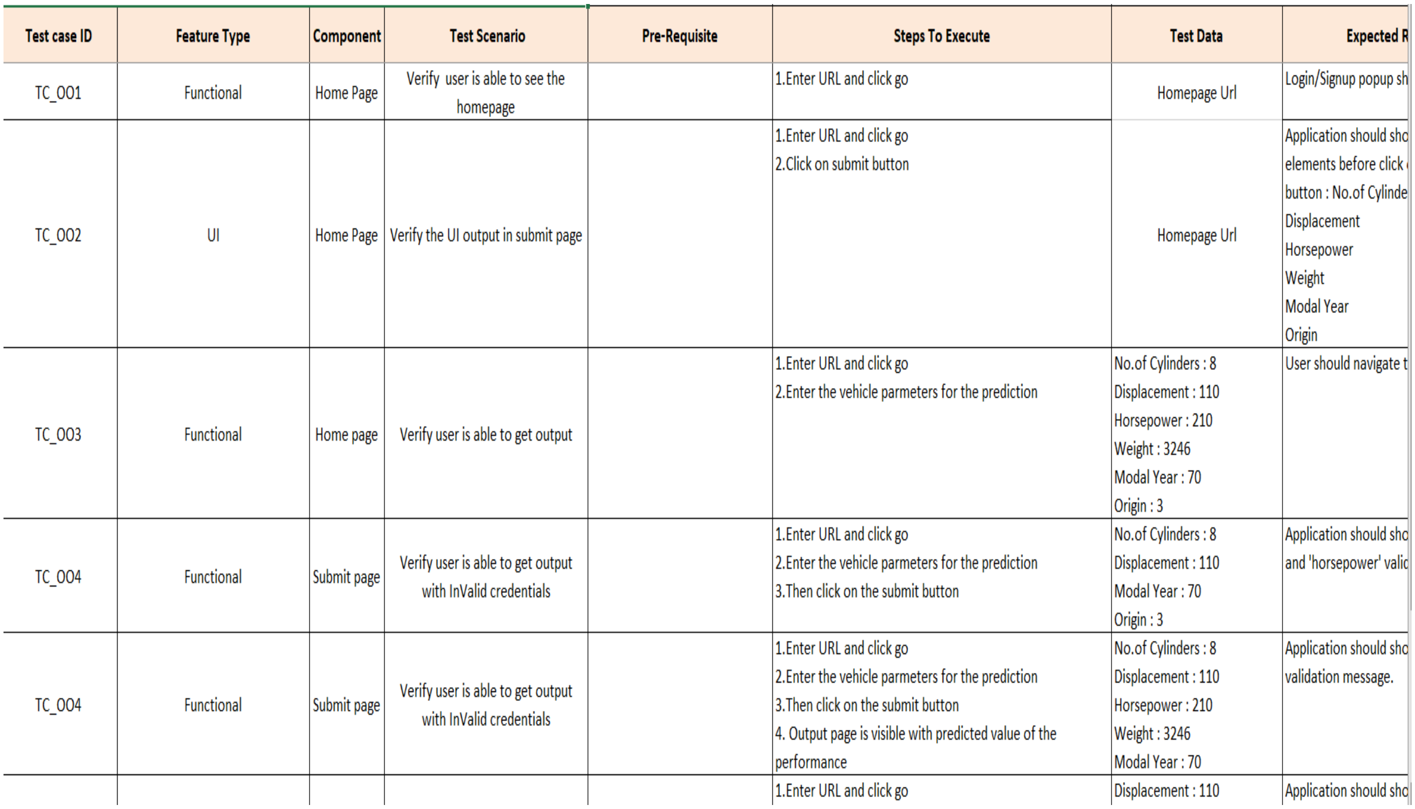
Average velocity of sprint-4: AV = 15/4 = 3.75

Burn down chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as scrum. However, burn down charts can be applied to any project containing measurable progress over time.



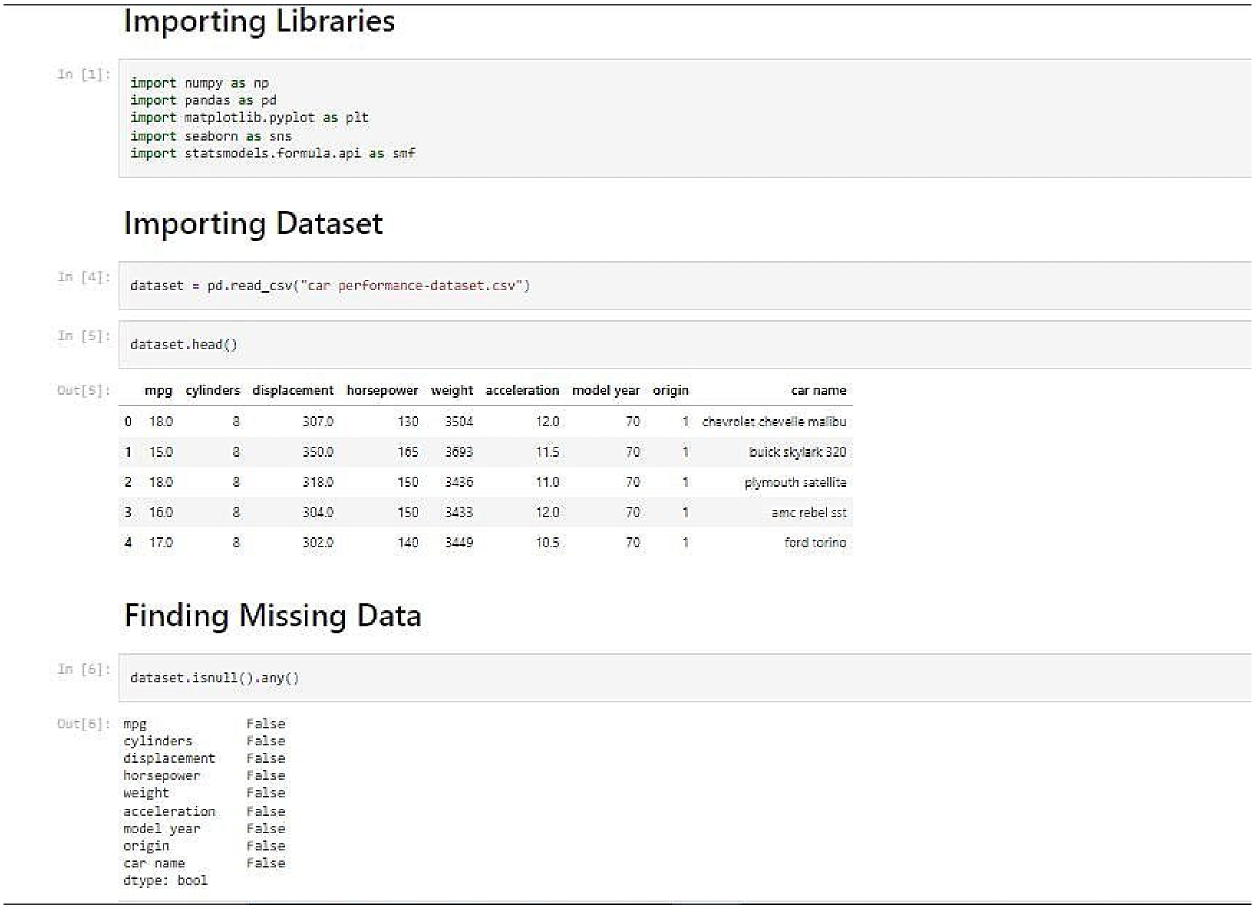
6.3 Reports from Jira



Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected R
TC_001	Functional	Home Page	Verify user is able to see the homepage		1.Enter URL and click go	Homepage Url	Login/Signup popup sh
TC_002	UI	Home Page	Verify the UI output in submit page		1.Enter URL and click go 2.Click on submit button	Homepage Url	Application should sho elements before click button : No.of Cylinde Displacement Horsepower Weight Modal Year Origin
TC_003	Functional	Home page	Verify user is able to get output		1.Enter URL and click go 2.Enter the vehicle parmeters for the prediction	No.of Cylinders : 8 Displacement : 110 Horsepower : 210 Weight : 3246 Modal Year : 70 Origin : 3	User should navigate t
TC_004	Functional	Submit page	Verify user is able to get output with InValid credentials		1.Enter URL and click go 2.Enter the vehicle parmeters for the prediction 3.Then click on the submit button	No.of Cylinders : 8 Displacement : 110 Modal Year : 70 Origin : 3	Application should sho and 'horsepower' valid
TC_004	Functional	Submit page	Verify user is able to get output with InValid credentials		1.Enter URL and click go 2.Enter the vehicle parmeters for the prediction 3.Then click on the submit button 4. Output page is visible with predicted value of the performance	No.of Cylinders : 8 Displacement : 110 Horsepower : 210 Weight : 3246 Modal Year : 70	Application should sho validation message.
					1.Enter URL and click go	Displacement : 110	Application should sho

7.Coding & Solutioning

Feature-1



Importing Libraries

```
In [1]:  import numpy as np
         import pandas as pd
         import matplotlib.pyplot as plt
         import seaborn as sns
         import statsmodels.formula.api as smf
```

Importing Dataset

```
In [4]:  dataset = pd.read_csv("car performance-dataset.csv")
```

```
In [5]:  dataset.head()
```

Out[5]:

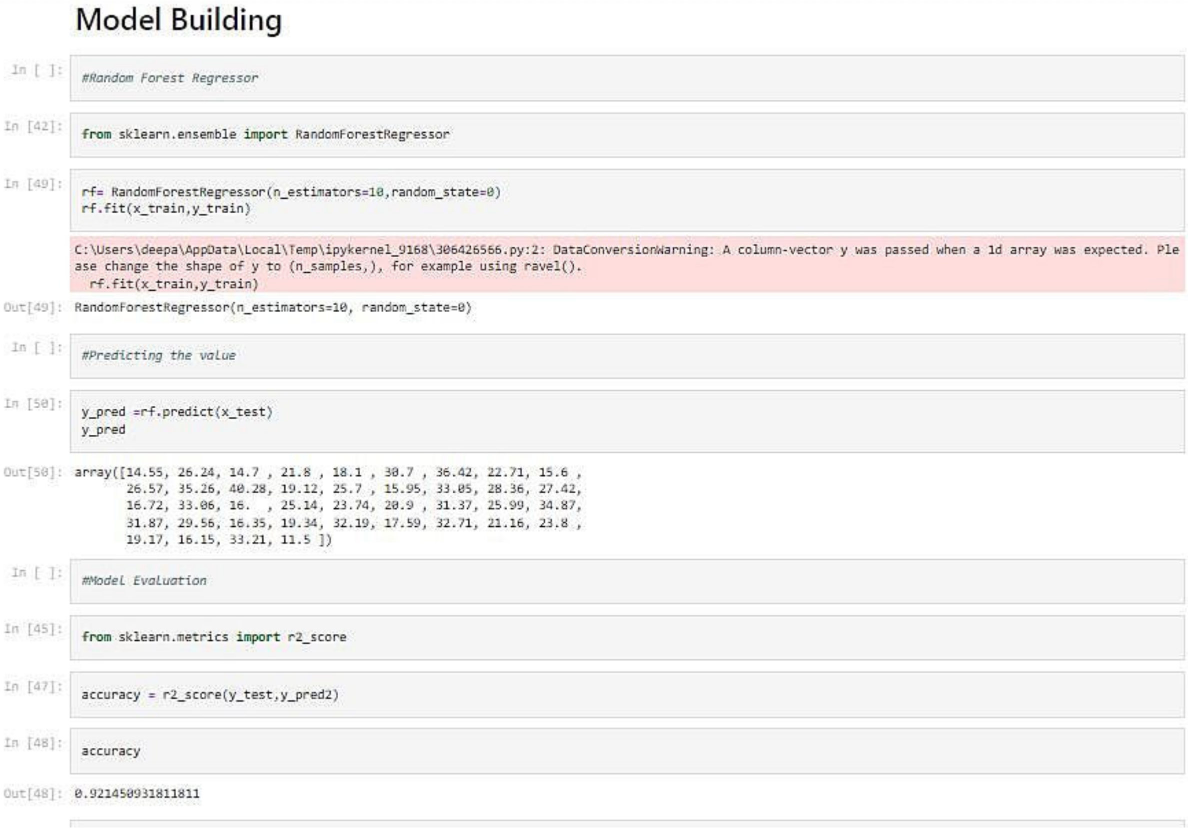
	mpg	cylinders	displacement	horsepower	weight	acceleration	model year	origin	car name
0	18.0	8	307.0	130	3504	12.0	70	1	chevrolet chevelle malibu
1	15.0	8	350.0	165	3693	11.5	70	1	buick skylark 320
2	18.0	8	318.0	150	3436	11.0	70	1	plymouth satellite
3	16.0	8	304.0	150	3433	12.0	70	1	amc rebel sst
4	17.0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

Finding Missing Data

```
In [6]:  dataset.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
```

Feature-2



Model Building

```
In [ ]:   #Random Forest Regressor

In [42]:  from sklearn.ensemble import RandomForestRegressor

In [49]:  rf= RandomForestRegressor(n_estimators=10,random_state=0)
          rf.fit(x_train,y_train)

          C:\Users\deepa\AppData\Local\Temp\ipykernel_9168\306426566.py:2: DataConversionWarning: A column-vector y was passed when a 1d array was expected. Ple
          ase change the shape of y to (n_samples,), for example using ravel().
            rf.fit(x_train,y_train)

Out[49]:  RandomForestRegressor(n_estimators=10, random_state=0)

In [ ]:   #Predicting the value

In [50]:  y_pred =rf.predict(x_test)
          y_pred

Out[50]:  array([14.55, 26.24, 14.7 , 21.8 , 18.1 , 30.7 , 36.42, 22.71, 15.6 ,
                 26.57, 35.26, 40.28, 19.12, 25.7 , 15.95, 33.05, 28.36, 27.42,
                 16.72, 33.06, 16.  , 25.14, 23.74, 20.9 , 31.37, 25.99, 34.87,
                 31.87, 29.56, 16.35, 19.34, 32.19, 17.59, 32.71, 21.16, 23.8 ,
                 19.17, 16.15, 33.21, 11.5 ])

In [ ]:   #Model Evaluation

In [45]:  from sklearn.metrics import r2_score

In [47]:  accuracy = r2_score(y_test,y_pred2)

In [48]:  accuracy

Out[48]:  0.921450931811811
```

TESTING REPORT

Testing of an individual software component or module is termed as Unit Testing. It is

typically done by the programmer and not by testers, as it requires detailed

knowledge of the
internal program design and code.

The Code was developed in 3 separate parts-

1. AI Model developed using Jupyter Notebook
2. Web Front end was developed using VS Code

UNIT TESTING:

Unit testing is carried out screen-wise, each screen being identified as an object. Attention is diverted to individual modules, independently to one another to locate errors .Thishas enabled the detection of errorsin coding and logic .Thisisthe first level of testing.In this, codes are written such thatfromone module ,we can move ontothenextmoduleaccordingtothechoiceweenter.

SYSTEM TESTING:

In this, the entire system was tested as a whole with all forms, code, modules and class modules .System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commences.

It is a series of different tests that verifies that all system elements have been properly integrated and perform allocated functions.

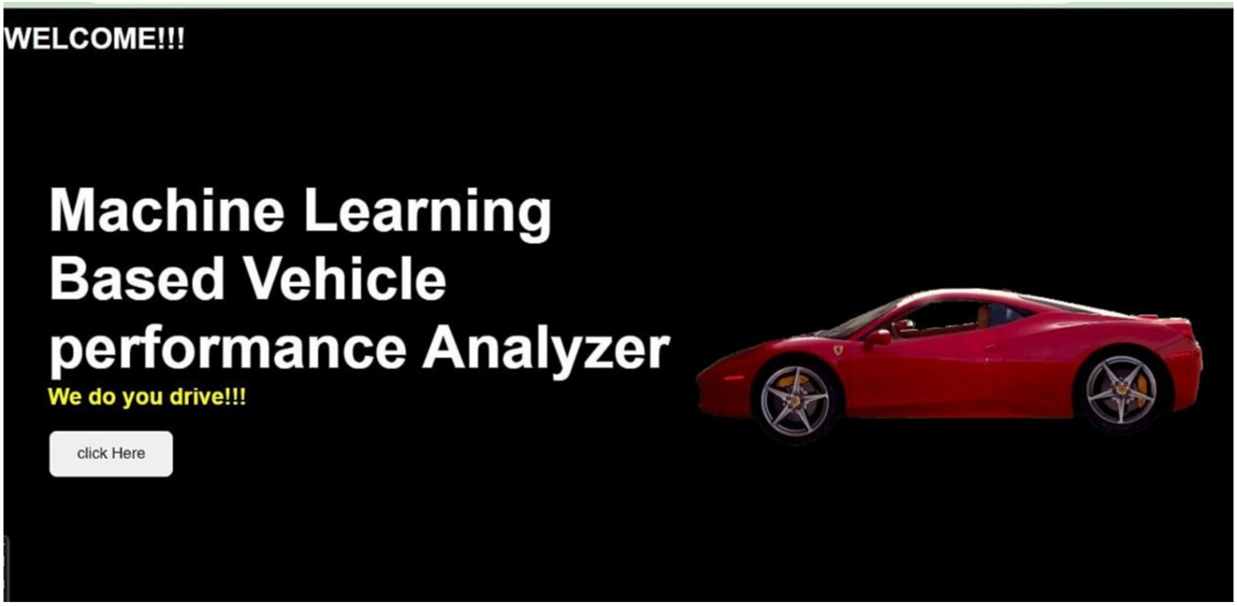
System testing makes logical assumptions that if all parts of the system are correct, the goal will be successfully achieved. Testing is the process of executing the program with the intent of finding errors.

Testing cannot show the absence of defects, it can only show that software errors are present.

8.1 Test cases :

TEST CASE	No of Cylinders	Displacement	HP	Weight	Year	Origin	Predicted Value

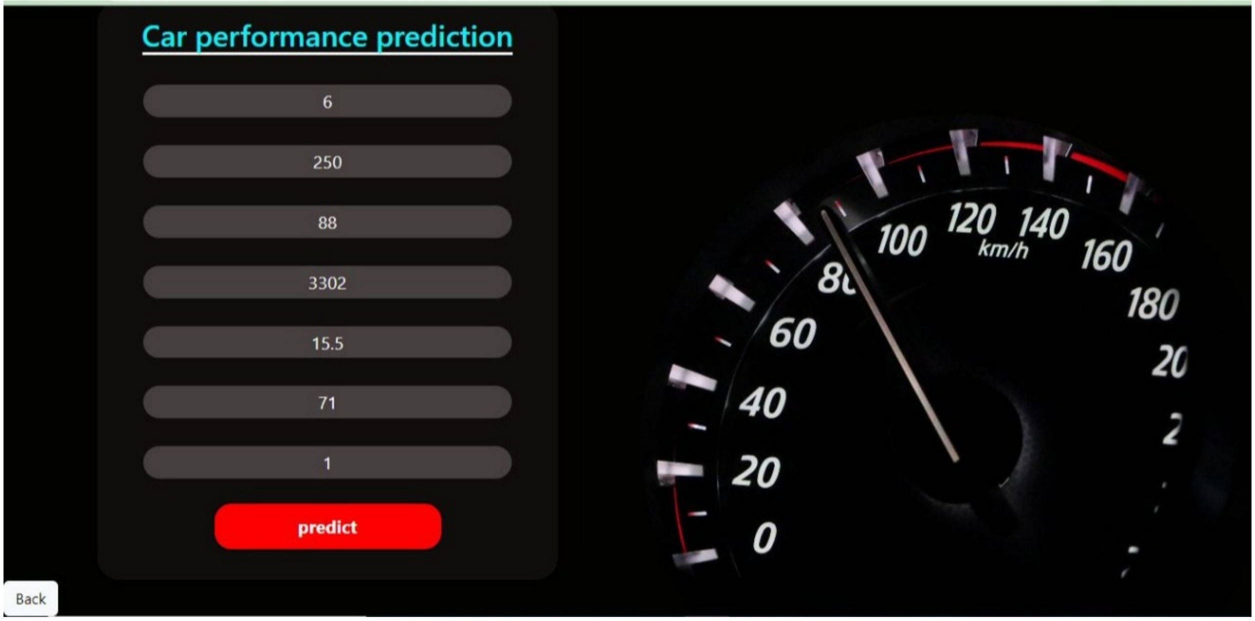
1	8	307	130	3504	70	1	18.1
2	8	350	165	3693	70	1	15.2
3	4	130	95	2372	70	3	24.2
4	6	198	95	2833	70	1	22.3
5	4	104	95	2375	70	2	24.2

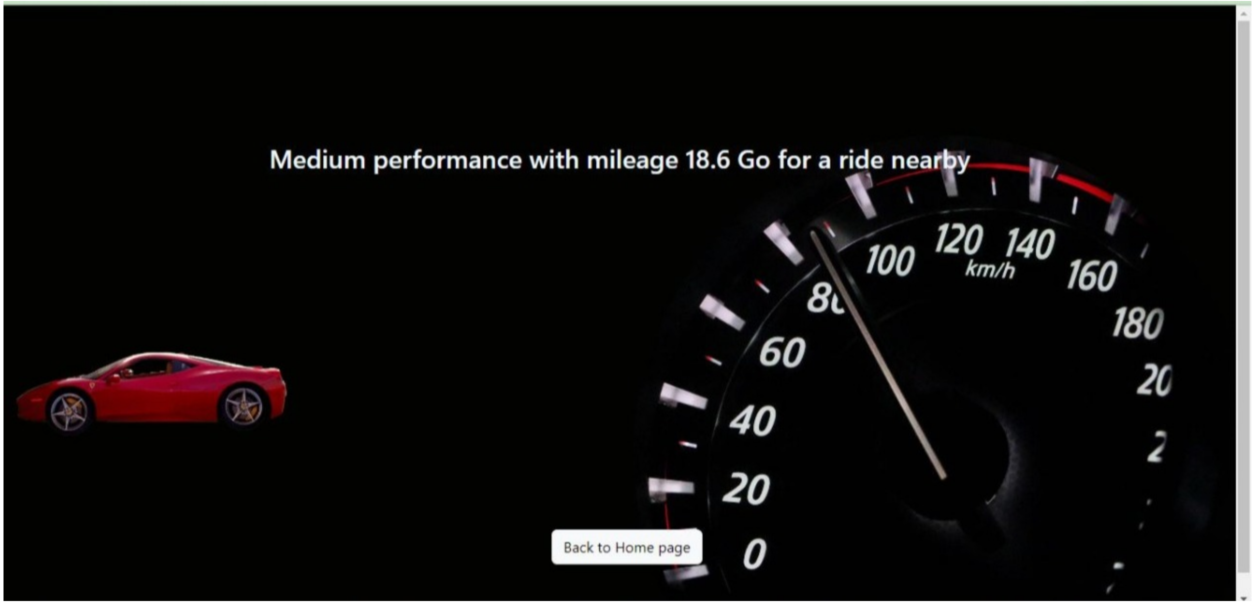


WELCOME!!!

Machine Learning Based Vehicle performance Analyzer

We do you drive!!!





Medium performance with mileage 18.6 Go for a ride nearby

8.2 User Acceptance Testing :

Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the Machine Learning-based Vehicle Performance Analyzer project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis.

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	9	0	0	9
Client Application	44	0	0	44
Security	2	0	0	2

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

1. Test Case Analysis

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	15	6	2	3	26
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	12	3	5	22	42
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	4	2	1	7
Totals	30	16	14	28	88

This report shows the number of test cases that have passed, failed, and untested

Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

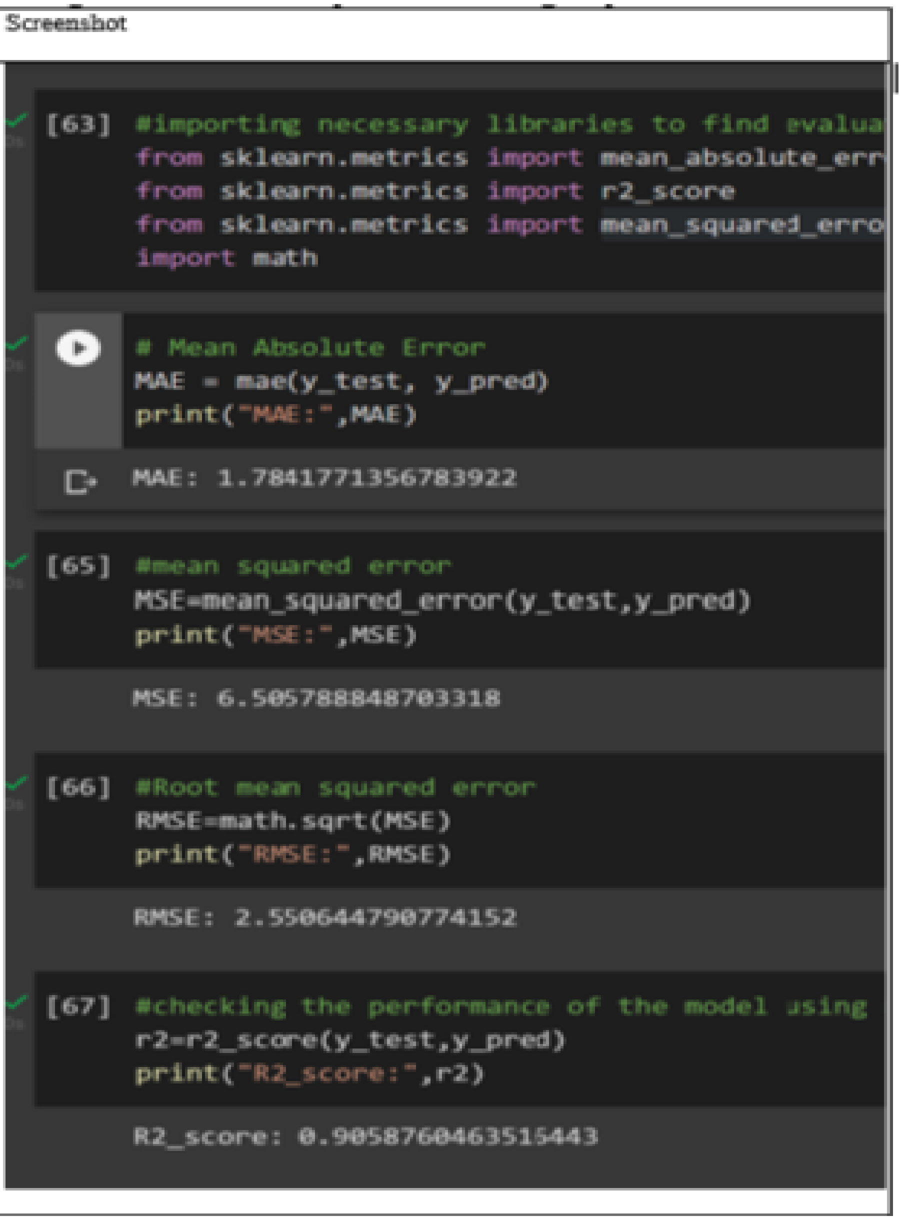
9 . Results :

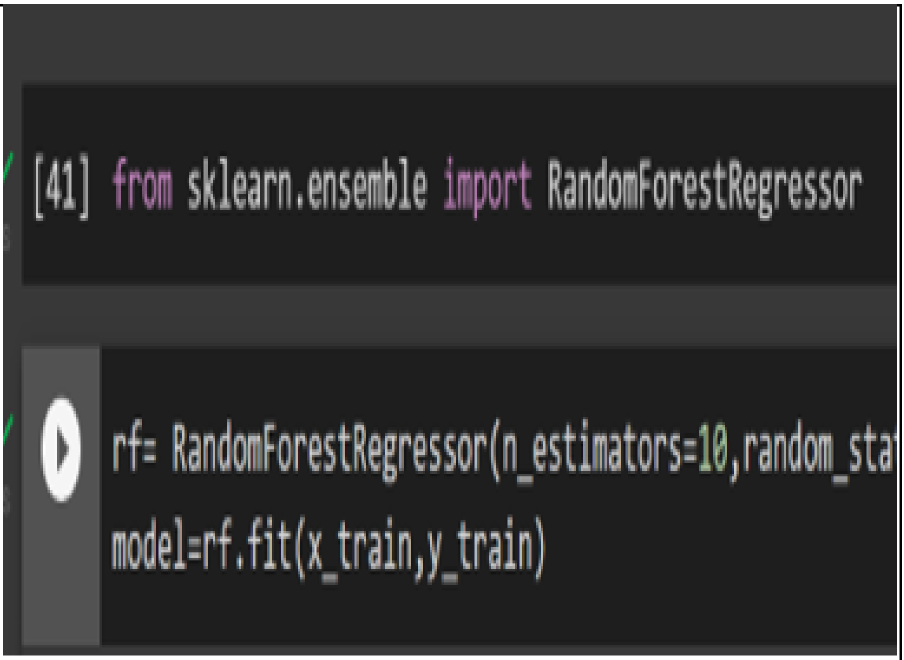
9.1 Performance Metrics :

Model Performance Testing:

Project team shall fill the following information in model performance testing template.

S.No.	Parameter	Values	Screenshot

1.	Metrics	Regression Model:
		MAE - 1.7841,
		MSE - 6.5057,
		RMSE - 2.5506 ,
		R2 score – 0.9058
			
			[63] #importing necessary libraries to find evalua
			from sklearn.metrics import mean_absolute_err
			from sklearn.metrics import r2_score
			from sklearn.metrics import mean_squared_erro
			import math
			# Mean Absolute Error
			MAE = mae(y_test, y_pred)
			print("MAE:",MAE)
			MAE: 1.7841771356783922
			[65] #mean squared error
			MSE=mean_squared_error(y_test,y_pred)
			print("MSE:",MSE)
			MSE: 6.505788848703318
			[66] #Root mean squared error
			RMSE=math.sqrt(MSE)
			print("RMSE:",RMSE)
			RMSE: 2.550644790774152
			[67] #checking the performance of the model using
			r2=r2_score(y_test,y_pred)
			print("R2_score:",r2)
			R2_score: 0.9058760463515443

2.	Tune the Model	Hyperparameter Tuning –	

```
[41] from sklearn.ensemble import RandomForestRegressor
```

```
rf= RandomForestRegressor(n_estimators=10,random_sta
model=rf.fit(x_train,y_train)
```

TEST CASE	No of Cylinders	Displacement	HP	Weight	Year	Origin	Predicted Value
1	4	120	97	2506	72	3	23
2	4	98	80	2164	72	1	28
3	4	97	88	2100	72	3	27
4	8	350	175	4100	73	1	13
5	8	304	150	3672	73	1	14

10. ADVANTAGES & DISADVANTAGES :

ADVANTAGES:

- It helps users for predicting the vehicle performance.
- Here the chance of occurrence of error is less when compared with the existing system.
- It is fast, efficient and reliable.
- Avoids data redundancy and inconsistency.
- Very user-friendly.
- Easy accessibility of data

DISADVANTAGES:

- computer literacy and network access
- Low Computer Literacy
- Security Concerns
- Authenticity
- Infrastructural Requirement.

11 . Conclusion :

The monitoring of car performance, especially gas consumption, has so far been approached only very superficially. A typical fuel gauge, when closely monitored, shows an extremely non-linear relationship between needle movement and fuel consumption. Inaccuraciesoccur especially in the range of critical low fuel values of 5-10% or more. In the past, due to thislimitation, some luxury cars had an audible and flashing light alarm function to indicate a low fuel condition. These systems, which add to the existing fuel level, have no more accuracy than the fuel level monitor alone. In recent years, with the availability of computer techniques and reliable and less expensive computer equipment, a number of systems have been developed to provide somewhatmore accurate information about vehicle performance.

12. FUTURE SCOPE :

This merits exploratory methods based on actual failures to deduct likely failure modes. This thesis presents two methods for data mining the vehicle maintenance records and vehicle usage data to learn usage or wear patterns indicative of failures. This requires detailed maintenance records where the failure root cause can be deducted with accurate date or mileages of the repair.

Further, more wide-spread adoption of predictive maintenance calls for automatic and less human-resource demanding methods, e.g. unsupervised algorithms with lifelong learning. Such methods are easier to scale up and they can thus be ubiquitously applied since much of the modelling is automated and requires little or no human interaction.

Maintenance predictions can be enhanced by combining the deviations in onboard data with off-board data sources such as maintenance records and failure statistics. This is exclusive product knowledge, only accessible to the vehicle manufacturers, which gives them an advantage in predicting maintenance. Still, data mining has yet to become a core competence of vehicle manufacturers, which makes the road to industrialisation long. The aim of this thesis is to investigate how on-board data streams and off-board data can be used to predict the vehicle maintenance. More specifically, how on-board data streams can be represented and compressed into a transmittable size and still be relevant for maintenance predictions. Further, the most interesting deviations must be found for a given repair which requires new ways of combining semantic maintenance records with deviations based on compressed on-board data.

This can be accessed anytime anywhere, since it is a web application provided only an internet connection.

13. APPENDIX

app.py

```
from flask import request
from flask import Flask, render_template
from flask_cors import CORS
import joblib

app=Flask(__name__)

@app.route('/')
def index():
    return render_template("a.html")

@app.route('/another')
def another():
    return render_template('another.html')

@app.route('/about')
def about():
    return render_template('about.html')

@app.route('/contact')
def contact():
    return render_template('contact.html')

@app.route('/Register',methods=["POST","GET"])

def Register():
    if request.method=="POST":
        cylinders=request.form.get('12')
        Displacement=request.form.get('34')
        Power=request.form.get('56')
        Weight=request.form.get('3')
        model=request.form.get('4')
        origin=request.form.get('5')
        war=request.form.get('6')
        X=[[cylinders,Displacement,Power,Weight,model,origin,war]]
        model=joblib.load("Regression.pkl")
        car=model.predict(X)[0]


        if(car >46 ):
            return render_template("result.html",Efficiency = "Very High performance with mileage"+ str(car)+"  You can plan for a tour")
        elif(car >29 and car<=46):
            return render_template("result.html",Efficiency = "High performance with mileage    "+ str(car)+"   Go for a healthy ride")
        elif(car >17.5 and car<=29):
            return render_template("result.html",Efficiency = "Medium performance with mileage  " + str(car)+"  Go for a ride nearby")
        elif(car >9 and car<=17.5):
            return render_template("result.html",Efficiency = "Low performance with mileage     "+ str(car)+"   Don't go to a long distance")
        else:
            return render_template("result.html",Efficiency = "Worst performance with mileage   "+ str(car)+"   carry extra fuel")




if __name__=='__main__':
    app.run(debug=True)
```

app_ibm.py

```
import requests
import flask
from flask import request, render_template,Flask
from flask_cors import CORS
import requests



API_KEY = "1LuQwWCcDkqNEAHl6QzF6F5zfSDv8Az4s0HLpzHirICX"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token',
data={"apikey":
 API_KEY, "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}



app=flask.Flask(__name__)

@app.route('/')
def index():
    return render_template("a.html")

@app.route('/another')
def another():
    return render_template('another.html')

@app.route('/about')
def about():
    return render_template('about.html')

@app.route('/contact')
```

```
def contact():
    return render_template('contact.html')

@app.route('/Register',methods=['POST'])

def register():

    cyl=float(request.form['12'])
    dis=float(request.form['34'])
    hp=float(request.form['56'])
    w=float(request.form['3'])
    a=float(request.form['4'])
    my=float(request.form['5'])
    ori=float(request.form['6'])
    X = [[cyl, dis, hp, w, a, my, ori]]
    payload_scoring = {"input_data": [{"field": [['12', '34', '56', '3', '4', '5', '6']],
"values": X}]}
    response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/b15feb78-c8da-49a8-9d5d-
ac409a0470eb/predictions?version=2022-11-19',
    json=payload_scoring, headers={'Authorization': 'Bearer ' + mltoken})
    print(response_scoring)
    predictions = response_scoring.json()
    predict = predictions['predictions'][0]['values'][0][0]
    print("Final prediction :", predict)
    if(predict >46 ):
        return render_template("result.html",Efficiency = "Very High performance
with mileage"+ str(predict)+"  You can plan for a tour")
    elif(predict >29 and predict<=46):
        return render_template("result.html",Efficiency = "High performance with
mileage    "+ str(predict)+"   Go for a healthy ride")
    elif(predict >17.5 and predict<=29):
        return render_template("result.html",Efficiency = "Medium performance with
mileage  " + str(predict)+"  Go for a ride nearby")
    elif(predict >9 and predict<=17.5):
        return render_template("result.html",Efficiency = "Low performance with
mileage     "+ str(predict)+"   Don't go to a long distance")
    else:
        return render_template("result.html",Efficiency = "Worst performance with
mileage   "+ str(predict)+"   carry extra fuel")

if __name__=='__main__':
    app.run(debug=True)
```

a.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Vehicle performance Analyzer</title>
            <link rel="stylesheet" href="{{url_for('static',filename='css/new.css')}}">
</head>
<body>
    <header>
        <h1>WELCOME!!!</h1>
         <nav>
            <!-- <li>Home</li>
            <li>About</li>
            <li>Contact us</li> -->
        </nav>
    </header>
    <div class="main">
        <div class="content">

            <h1>Machine Learning Based Vehicle performance Analyzer</h1>
            <h3><span>We do you drive!!!</span></h3>

                                  <form action="\another">
                                                         <button >click Here</button>
                                  </form>

        </div>
        <div class="image">
                                  <image
src="{{url_for('static',filename='img/car.gif')}}"style="width:600px" >
        </div>
    </div>

</body>
</html>
```

another.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Prediction page</title>
    <link rel="stylesheet"
href="{{url_for('static',filename='css/bootstrap.min.css')}}">
```

```
    <link rel="stylesheet" href="{{url_for('static',filename='css/style.css')}}">
    <style>
         input[type="text"]{
             text-align: center;
            }
        body
        {
                height:90vh;
                width: 100%;
                /* background: linear-gradient(black,rgb(44, 41, 41),black);
                background-size: cover; */

            background-size:cover;
            background-image: url({{url_for('static',filename='img/dark.jpg')}})
        }
    </style>
</head>
<body>
    <form action="Register" method="post" class="form" autocomplete="off">

        <h1>Car performance prediction</h1>
        <input type="text" name="12" class="box" placeholder="No. of cylinders">
        <input type="text" name="34" class="box" placeholder="Displacement">
        <input type="text" name="56" class="box" placeholder="Horse Power">
        <input type="text" name="3" class="box" placeholder="Weight in pounds">
        <input type="text" name="4" class="box" placeholder="Acceleration">
        <input type="text" name="5" class="box" placeholder="Model year">
        <input type="text" name="6" class="box" placeholder="Origin">
        <input type="submit" value="predict" id="submit">
    </form>
    <form action="\">
        <input type="submit" value="Back" class="btn btn-light">
```

```
            </form>

</body>
</html>
```

result.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
    <link rel="stylesheet"
href="{{url_for('static',filename='css/bootstrap.min.css')}}">
    <link rel="stylesheet" href="{{url_for('static',filename='css/result.css')}}">
    <style>
         input[type="text"]{
             text-align: center;
            }
        body
        {
                height:90vh;
                width: 100%;
                /* background: linear-gradient(black,rgb(44, 41, 41),black);
                background-size: cover; */

            background-size:cover;
            background-image: url({{url_for('static',filename='img/dark.jpg')}})
        }
    </style>
</head>
```

```
<body>
    <div>
        <h3>{{Efficiency}}</h3>
    </div>
     <div class="image">
                                    <image src="{{url_for('static',filename='img/car.gif')}}" >
        </div>


    <form action="\">
            <input type="submit" value="Back to Home page" class="btn btn-light">
    </form>

</body>
</html>
```

new.css

```
*{
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    cursor: pointer;
    font-family: sans-serif;
}
body{
    width: 100%;
    min-height: 90vh;
    background: black;
}
header{
    display:flex;
    justify-content: space-between;
    padding: 15px 20px;
    color: white;
    animation: header 2s ease-in-out ;
}
@keyframes header{
    0%{
        transform: translateY(-100%);
    }

}
nav li{
    display:inline-block;
    margin: 10px 20px;
    font-weight: 600;
}
.main{
    width: 100%;
    display:flex;

}
.content{
    width: 50%;
    margin-top: 8%;
    margin-left: 5%;
    color: white;
    animation: content 2s ease-in-out;

}
```

```
@keyframes content{
   0%{
      opacity: 0;
      transform: translateX(-50%);
   }
}
.content h1{
   font-size: 65px;
   margin-top: 1%;

}
.content h3{
   font-size: 25px;
   margin-bottom: 3%;
   font-family: Verdana, Geneva, Tahoma, sans-serif;
}
span{

   color: yellow;
   margin-bottom: 3%;
}
button{
   padding: 15px 30px;
   background: transparent rgb(22, 21, 21);
   font-size: 17px;
   border-radius: 10px;
}
button:hover{
   background: rgb(255, 10, 10);
}
.image{
   padding-top: 170px;
}
```

style.css

```
/*box*/
.form{

    display: flex;
    flex-direction: column;
    height:625px;
    width: 500px;
    border:0.5px solid black;
    align-items:center;
    margin:auto;
    margin-top: 20px;
    margin-left: 100px;
    background-color: rgba(151, 137, 137, 0.089);
    box-shadow: inset .5px .5px rgba(0,0,0, 0.4);
    border-radius: 25px;

}
.form h1{

    color: rgb(14, 240, 248);
    font-size: 2rem;
    border-bottom: 3px solid rgba(255,255,255.05);
    margin: 17px;
}

.box{
    padding: 3px;
    margin: 15px;
    width: 80%;
    border: none;
    outline: none;
    border-radius: 20px;
    background-color: rgba(165, 155, 155, 0.356);
    color:rgb(253, 250, 250);
    font-size: 1.2rem;

}

#submit{
    padding: 10px 20px;
    margin-top: 10px;
    width: 50%;
    background-color:red;
    color:white;
    font-size: 1.2rem;
    font-weight: bold;
    border-radius: 20px;



}
#submit:hover{
    cursor: pointer;
    background-color: rgba(255,255,255.03);
    color: rgb(19, 37, 35);
}
::placeholder{
    color: white;
}
```

result.css

```
h1{
    height: 150px;

    text-align: center;
}

h3
{
    padding: 150px 100px;
    text-align: center;
    color: aliceblue;
}
form{
    padding: 70px 600px;

}
/* .image
{
    width: 35%;
    margin-top: 8%;
    z-index:-1;

}  */
img{
    width: 20rem;
    animation: car 5s ease-in-out;
    z-index:-5
}
@keyframes car{
    0%{
        transform: translateX(100%);
    }
    100%{
        transform: translateX(-220%);
    }
}
@media screen and (max-width:1200px) {
    .content{
        width: 100%;
        text-align: end;
    }
    .image{
        width: 100%;
        position: absolute;
        bottom: -5%;
        display: flex;
        justify-content:end;
    }
    img{
        width: 20rem;
    }

}
@media screen and (max-width:576px) {
    .content h1{
        margin-bottom: 5%;
        width: 100%;
        text-align: end;
    }
    .content h3{
        margin-bottom: 5%;
    }
    .image{
        width: 100%;
        position:absolute;
        bottom: -5%;
        display: flex;
        justify-content: end;
    }
    img{
        width: 20rem;
    }

}
```

GITHUB LINK : https://github.com/IBM-EPBL/IBM-Project-54094-1661594311

DEMO LINK