MACHINE LEARNING BASED VEHICLE PERFORMANCE ANALYZER

A PROJECT REPORT

Submitted by

ADHITYA RAJ R. 2019115008

AKASH D. 2019115010

KABILAN HUSSAIN M.K. 2019115044

NAVEEN PRAKAASHAM K.S.V. 2019115059

Team ID: PNT2022TMID35529

Contents

1. INTRODUCTION

1.1. Project Overview

The ability to estimate a car's performance level presents a big and fascinating challenge. The main goal of the current study is to forecast car performance in order to improve particular vehicle behavior. This can increase the system's efficiency and significantly lower the amount of gasoline it uses. performance evaluation of the car taking into account its horsepower, cylinder count, fuel type, and engine type, among other things. Based on these factors, the health of the car might be forecast. The process of gathering, looking into, evaluating, and documenting health data based on the three parts is ongoing.

Performance parameters like mileage, reliability, flexibility, and cost that may be combined are frequently used by both prediction engines and engine management systems. It is essential to analyze the elements using a number of well-known machine learning approaches, such as linear regression, decision trees, and random forests, in order to optimize the performance efficiency of the vehicle. The power, longevity, and range of automobile traction batteries are now the "hot topics" in automotive engineering. In this case, we additionally consider mileage performance. To answer this problem, we will build the models using a variety of methods and neural networks. We'll then compare which algorithm is most accurate in forecasting car performance (Mileage).

1.2. Purpose

Machine Learning models are used in finding the usage patterns of drivers and more applications based on input data obtained from sensors used in automobiles. Prediction of the performance of the automobile, in this case, cars, based on inputs obtained from working cars forms the major purpose of this project.

2. LITERATURE SURVEY

2.1. Existing Problem

The introduction of new technologies in recent years has enhanced the possibilities for processing data from automotive sensors. This kind of information is crucial, for example, when examining how drivers act while seated at the wheel. It hasn't been fully investigated by taking into account all of the sensors in a car engine because very little has been done to analyze driving patterns using data from car engine sensors. To close this gap, distributed online sensing platforms will be used to do classification and driver usage pattern discovery using machine learning techniques (supervised and unsupervised) on data from vehicle engine sensors**.**

2.2. References

Artificial Intelligence for Vehicle-to-Everything: A Survey

Wang Tong et. al in their paper titled "Artificial Intelligence for Vehicle-to-Everything: A Survey" have presented a comprehensive survey of the research works that have utilized AI to address various research challenges in Vehicle-to-Everything(V2X) systems and have summarized the contribution of those research works and categorized them according to the application domains, thereby presenting open problems and research challenges that need to be addressed for realizing the full potential of AI and Machine Learning in vehicular systems.[1]

Simulation for prediction of vehicle efficiency, performance, range and lifetime: A review of current techniques and their applicability to current and future testing standards

In the paper "Simulation for prediction of vehicle efficiency, performance, range and lifetime: A review of current techniques and their applicability to current and future testing standards", Abbas Fotouhi et. al. have developed on earlier work, exploring conventional and 'backward' techniques in the context of current NEDCbased UNECE vehicle testing standards and the proposed replacements based on the World Light Test Procedure. Model sensitivities for A, C, and D-segment vehicles are quantified and this is used to explore aspects where accurate models are key and where lower-fidelity representative models are appropriate. The paper also explores the sensitivity of predictions to PID control driver models and discusses the effect of cycle-following tolerance on

predictions. Finally, the paper proposes new standards - suitable for simulation or real-world testing - for a common quantification of in-use battery lifetime. [2]

Intelligent Performance Analysis of Automated Steering Systems for Autonomous Vehicles

The results of the research made by S. Salih and R. Olawoyin titled "Intelligent Performance Analysis of Automated Steering Systems for Autonomous Vehicles" showed that the proposed network control system had trained and validated more than 96.5% steering system behavior patterns and adapted large random disturbances of the steering controller commands. It is, therefore, necessary to develop artificial intelligence methodologies in automated steering systems of autonomous vehicles with neural networks representing the main topology blocks of the control system architecture and utilize ANN abstraction in the control system of autonomous vehicle steering control systems. The study's findings demonstrated that the suggested network control system had successfully trained and validated more than 96.5% of steering system behavior patterns and had successfully responded to significant random disturbances of steering controller commands. Therefore, it is essential to design artificial intelligence techniques for automated steering systems of autonomous vehicles that use ANN abstraction in the control system of the steering control system and neural networks as the basic topology building blocks of the control system architecture.[3]

Machine learning based real-time vehicle data analysis for safe driving modeling

In the paper “Machine learning based real-time vehicle data analysis for safe driving modeling“ the authors Pamul Yadav, Sangsu Sung and Dhananjay Singh say that the need to assess vehicle Meta characteristics that can assist drivers become more adept at avoiding collisions. It also highlights the need to assess how the quality of cars has changed over time. In this study, the driver's safety metrics and economical driving metrics are estimated using a linear regression model that is supervised learning-based. Over the course of a month, the data was gathered from fifteen separate drivers, totaling over 15,000 data points. And the criteria we developed could be used in automotive technology analysis to create cutting-edge intelligent automobiles. In the end, we examined the

parameter correctness over 80% for the driver's safety solution in a real world scenario and presented it.[4]

Modeling and Performance Analysis of Battery Enabled Vehicle

In the paper “Modeling and Performance Analysis of Battery Enabled Vehicle”, the authors Salil Patwardhan, R Bindu, Sushil Thale have discussed the significant problems of car emissions have been a significant problem. Research in the automotive industry has advanced, and the greatest solution found for addressing this issue is encouraging the use of electric vehicles, particularly battery electric vehicles. In order to do a thorough investigation before creating a real-time model, EV modeling is helpful. To ensure that the battery continues to operate in a safe manner, fuzzy logic control is employed to determine the regenerative braking component. The urban and highway driving cycles are taken into consideration separately when examining the significance of driving behavior. Variations in the vehicle's weight and in the road's rolling resistance, incline, and driving style are analyzed for their impacts.[5]

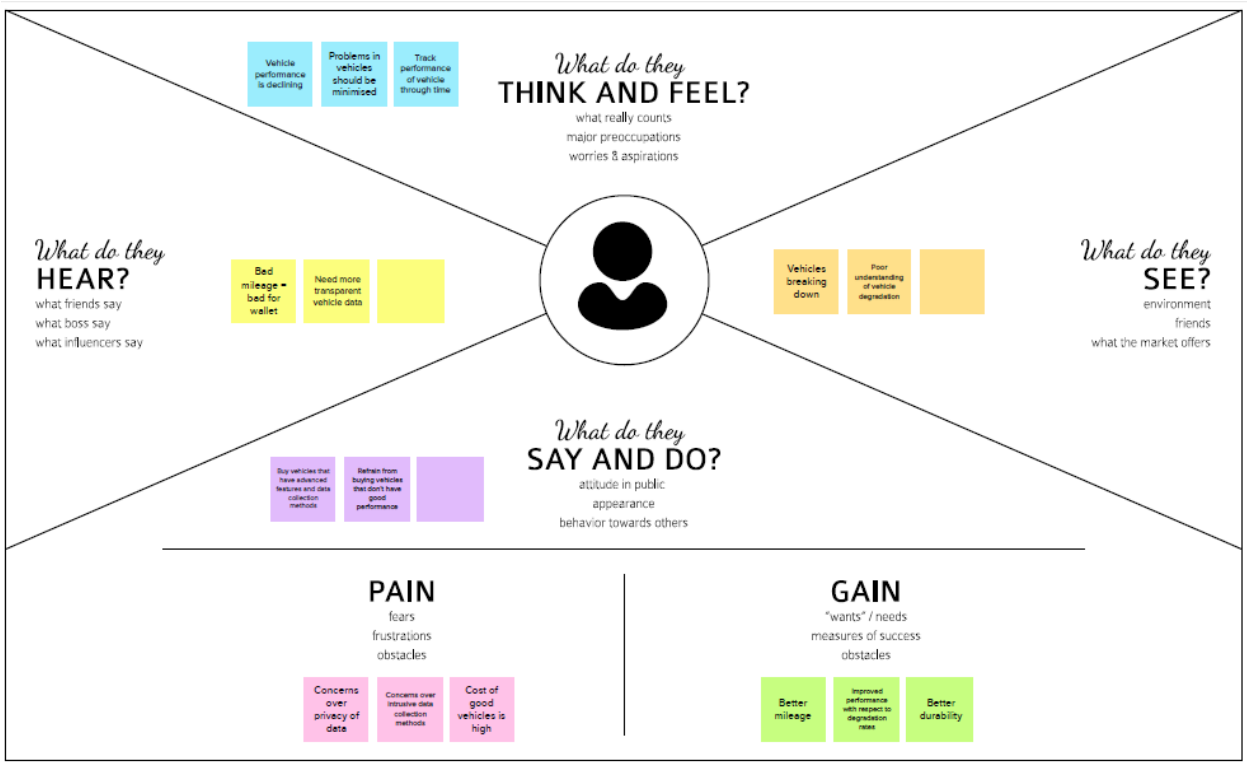
2.3. Problem Statement Definition

Problem Statement (PS)	I am (Customer)	I’m trying to	But	Because	Which makes me feel
PS-1	A user with an old car	Determine if my car’s mileage is performing as expected	Have no real answer apart from few online anecdotes	There is no way to find my car's specific model mileage	Concerned if I am spending more money which could have been saved

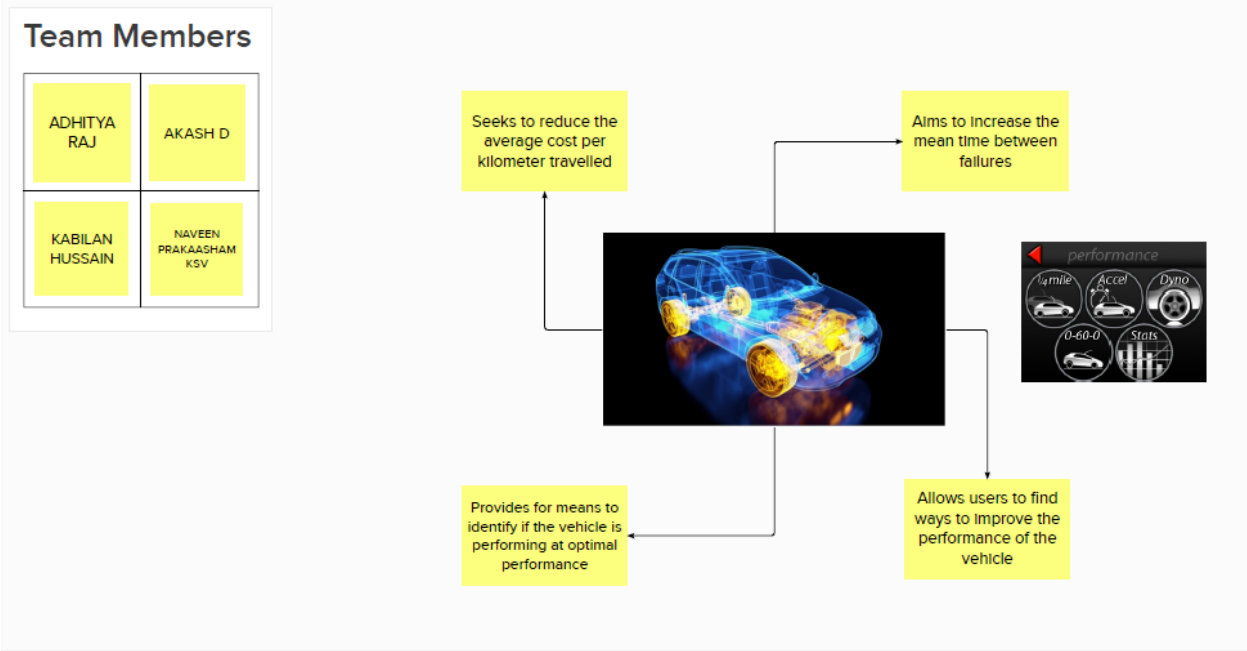
PS-2	Someone looking to replace my old car	Find cars that are similar to my existing one	Confused choosing between the variety of cars available in the market	They have similar features in the surface level	Confused and Frustrated over selecting a car

3. IDEATION & PROPOSED SOLUTION

3.1. Empathy Map Canvas



3.2. Ideation & Brainstorming



3.3. Proposed Solution

S. No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Predicting the performance level of a vehicle is an important and interesting problem. The main goal is to predict the performance of the vehicle to improve certain behaviors of the vehicle. This can significantly help to improve the system's fuel consumption and increase efficiency.
2.	Idea / Solution description	The performance analysis of the vehicle 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.
3.	Novelty / Uniqueness	We give the option to users to input their preferred choice of vehicle. If the data is available on the internet, web scraping is done to fetch appropriate feature values and then with these inputs the model will give the performance efficiency as the output.

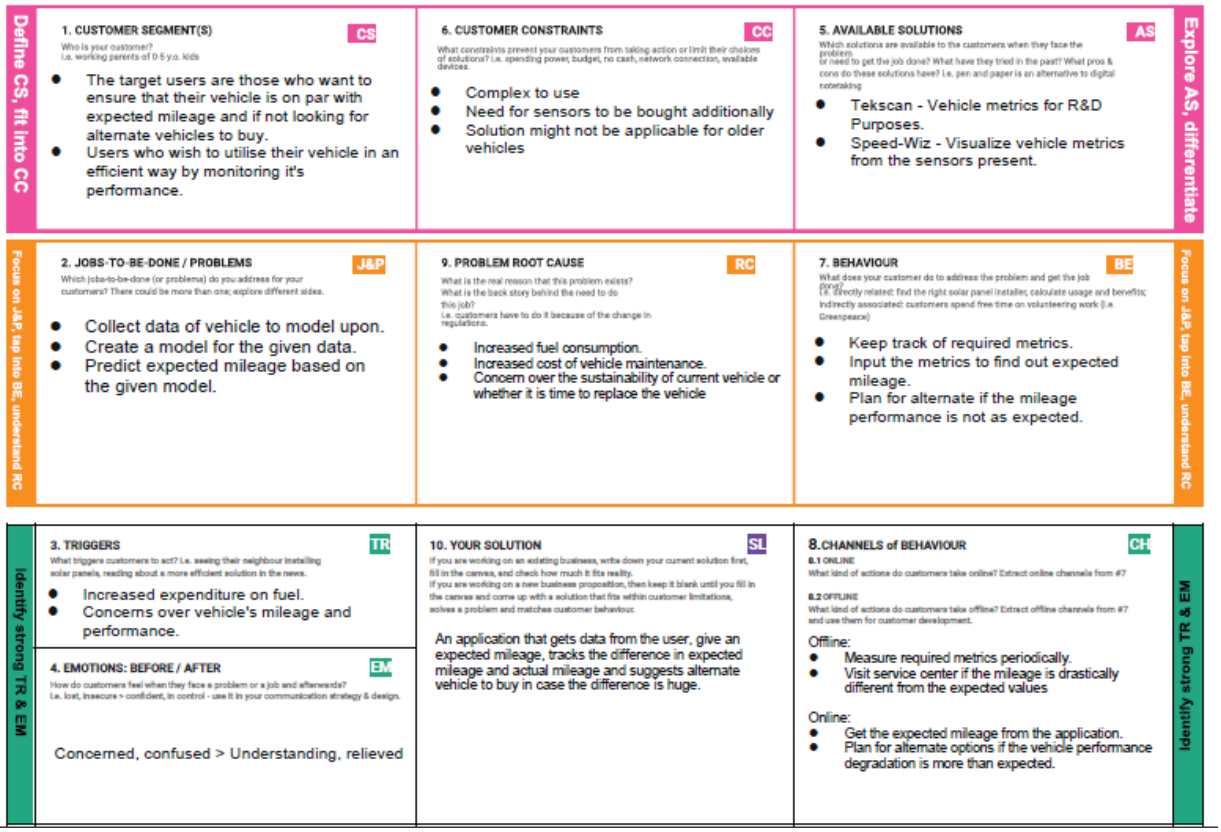
4.	Social Impact / Customer Satisfaction	Can help the customers by reducing their expenses on services and improve the life of the vehicle and its important parts. Can help the environment by reducing the fuel consumption, avoid air pollution and prevent wastage of resources by suggesting methods to use the vehicle efficiently.
5.	Business Model (Revenue Model)	Create a monthly subscription-based model of revenue that allows the user access to all or partial features based on the type of user account (basic/premium). Connect dealers with the consumer through the service for parts and repairs and charge a commission for each interaction.
6.	Scalability of the Solution	The solution could be done in a federated learning model which offloads most of the computation to the edge solutions, reducing the need for massive data storage, also decreasing the role of the server to gradient aggregation and averaging.

3.4. Problem Solution fit

Project Title: Machine Learning based Vehicle Performance Analyzer

Project Design Phase-I - Solution Fit Template

Team ID: ~~PNT2022TMIDxxxxxx~~ PNT2022TMID35529



4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Google O-Auth
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Vehicle Data Collection	User input through a Form Sending the data to the server
FR-4	Query Processing	Predict the expected mileage using the ML model Look for newer cars that are similar to the current model.
FR-5	Report Generation	Show the expected mileage, graph the expected mileage throughout time. Suggest similar car models from the database.

4.2 Non-Functional requirements

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

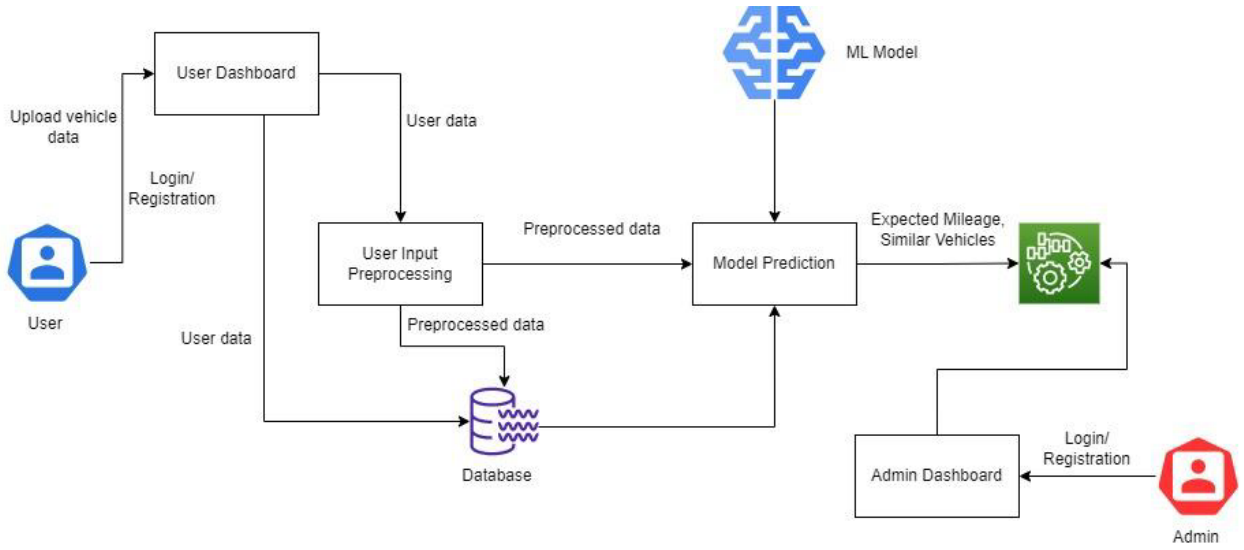
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	This application does not require any new specialized hardware to collect data. It tries to estimate mileage through data that the user can collect manually.
NFR-2	Security	Protected against all forms of web-based threats including but not limited to the OWASP Top 10

		vulnerabilities, ensuring Confidentiality, Integrity and Availability (CIA Triad).
NFR-3	Reliability	The application will give near perfect predictions regarding the efficiency and the remaining life span of the car and it will be devised such that the false positives will not affect the users badly in any way.
NFR-4	Performance	This application can support a reasonably large number of users accessing the services simultaneously, with little to no noticeable effect on the performance.
NFR-5	Availability	Ensuring that the application would be available to all the users at all the time, minimizing the downtime of the services.
NFR-6	Scalability	This application can be extended for all the vehicles, not only for cars.

5. PROJECT DESIGN

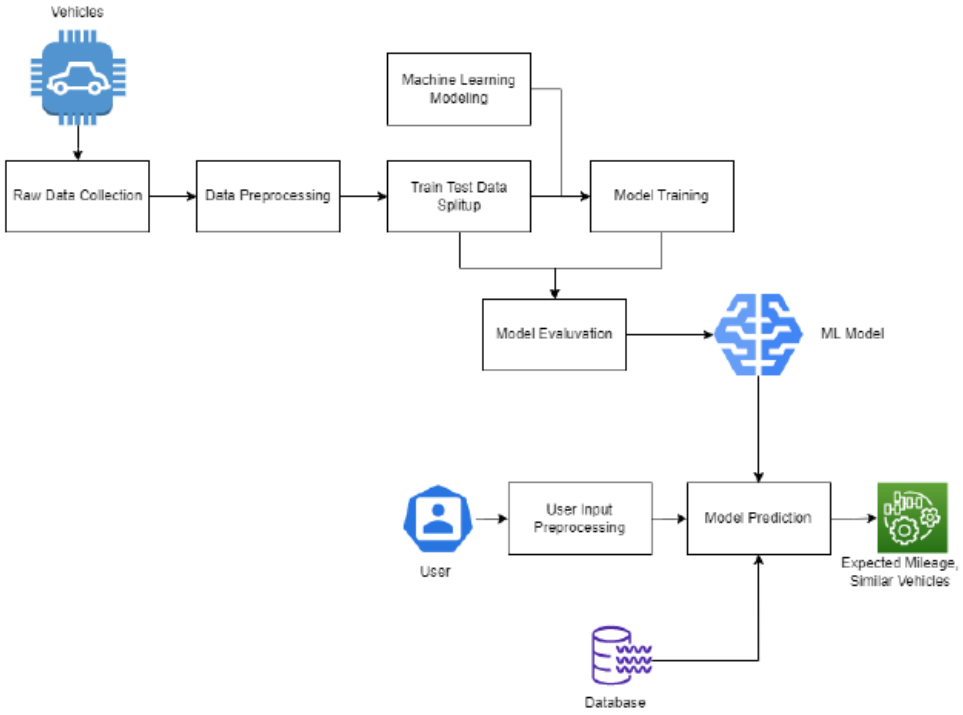
5.1 Data Flow Diagrams

The following is a data flow diagram depicting the flow of data between the different components in the application



5.2 Solution & Technical Architecture

The technical architecture of the project is given as follows



5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account /dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Google OAUTH		Medium	Sprint-1
	Login	USN-4	As a user, I can log into the application by entering email & password		High	Sprint-2
	Dashboard	USN-5	As a user, I can upload vehicle data and get an analysis of the data		Medium	Sprint-2
Admin	Login	USN-6	As an admin, I can log into the application by entering email & password	I can access my admin dashboard	High	Sprint-2
	Query Processing	USN-7	Predict the expected mileage using the ML model and look for newer cars that are similar to the current model.		Medium	Sprint-2

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset Collection		Collect the dataset containing the vehicle's information	10	High	Adhitya Raj Akash D Kabilan Naveen
Sprint-1	Dataset Preprocessing		Preprocess the dataset to get data ready to be trained with the machine learning model	5	High	Adhitya Raj Akash D Kabilan Naveen
Sprint-2	User Registration and Login	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	3	High	Adhitya Raj Akash D Kabilan Naveen
Sprint-2		USN-2	As a user, I can register for the application through form and Google O-auth	5	Low	Adhitya Raj Akash D Kabilan Naveen
Sprint-2		USN-3	As a user, I can log into the application by entering email & password	2	High	Adhitya Raj Akash D Kabilan Naveen

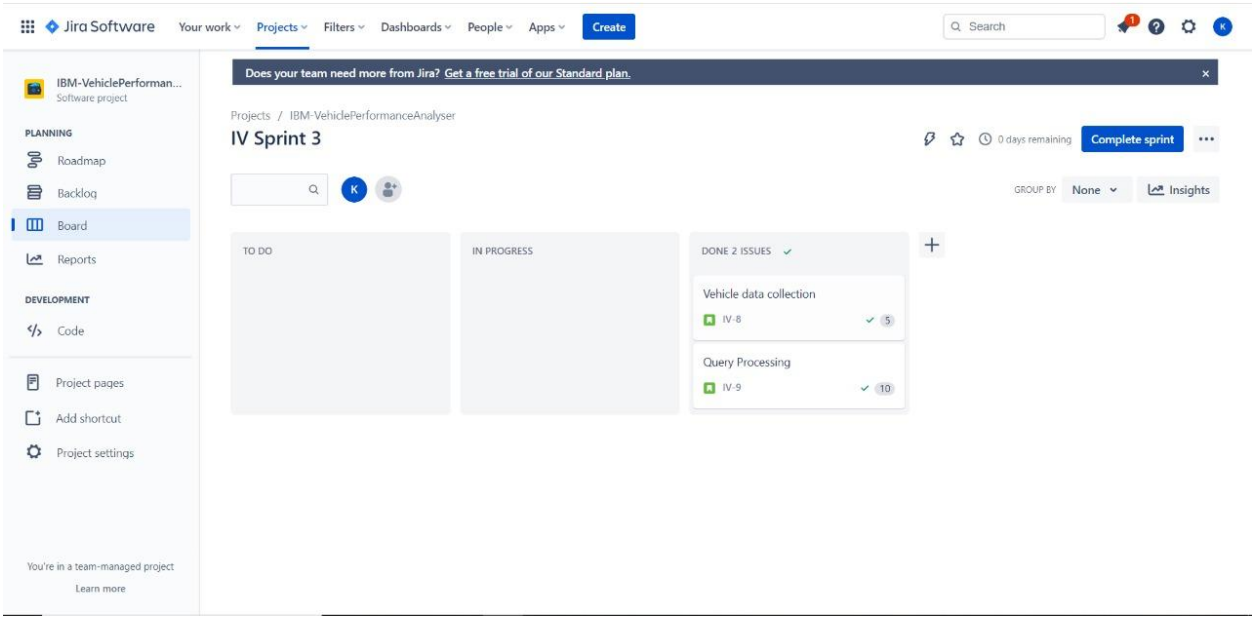
Sprint-2	User Confirmation	USN-4	As a user, I will receive confirmation email once I have registered for the application	3	High	Adhitya Raj Akash D Kabilan Naveen
Sprint-2	Model Building		ML Model building is done	7		Adhitya Raj Akash D Kabilan Naveen
Sprint-3	Vehicle Data Collection	USN-5	As a user, I can enter my inputs through a form in the application	2	Medium	Adhitya Raj Akash D Kabilan Naveen
Sprint-3		USN-6	As a user, I can send my input data to the Server	3	Medium	Adhitya Raj Akash D Kabilan Naveen
Sprint-3	Query Processing	USN-7	The system can predict the expected mileage using the ML model chosen	5	High	Adhitya Raj Akash D Kabilan Naveen
Sprint-3		USN-8	The system can look for newer cars that are similar to the current model	5	Low	Adhitya Raj Akash D Kabilan Naveen
Sprint-4	Report Generation	USN-9	The system can show the expected mileage, and graph the expected mileage throughout	7	High	Adhitya Raj Akash D Kabilan Naveen

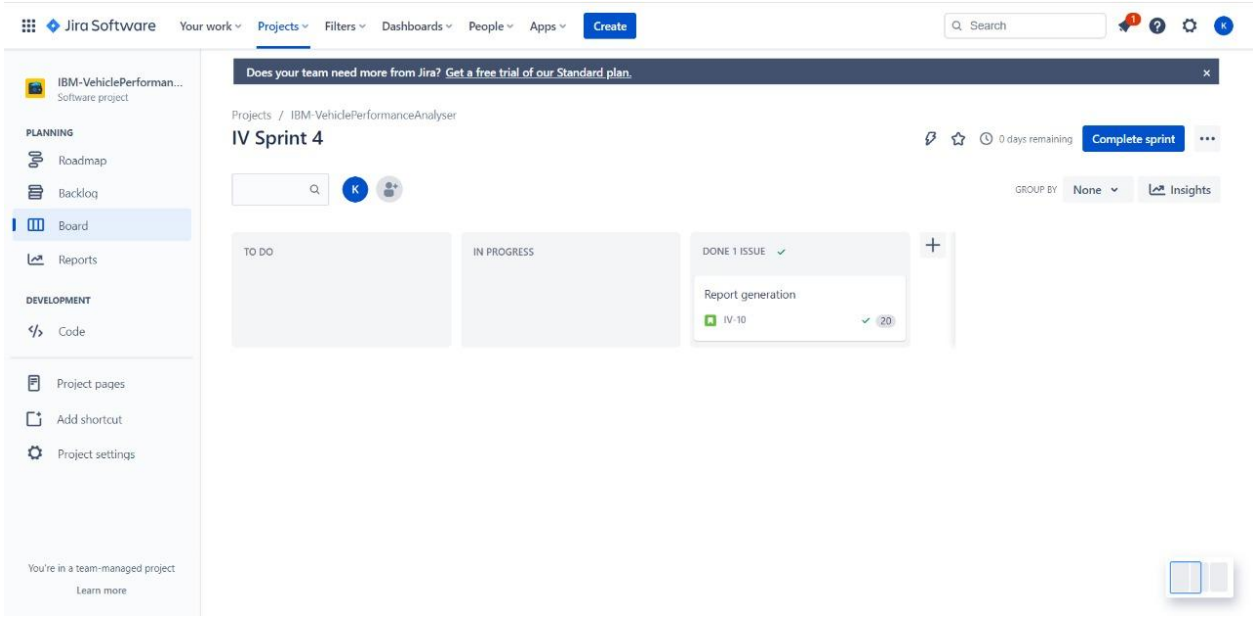
			time			
Sprint-4		USN-10	The system can suggest similar car models from the database	8	Low	Adhitya Raj Akash D Kabilan Naveen
Sprint-4		USN-11	As a user, I can view the Analysis reports of the vehicles and the suggested car models in the application	5	High	Adhitya Raj Akash D Kabilan Naveen

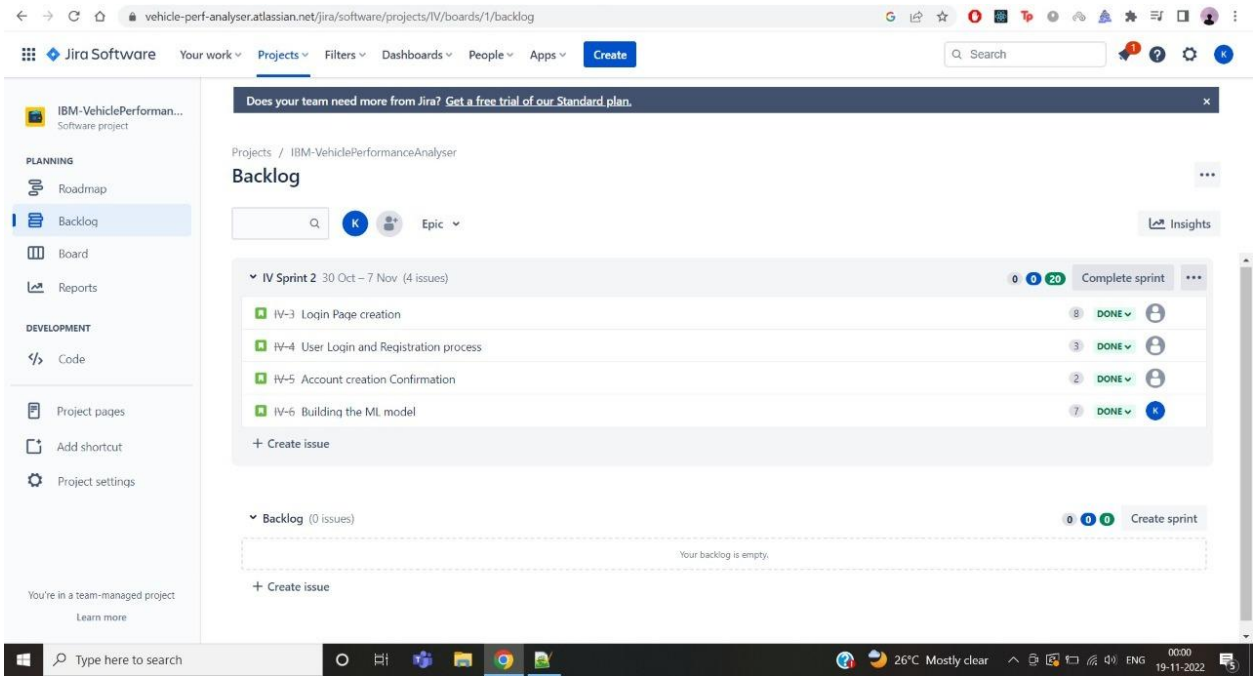
6.2 Sprint Delivery Schedule

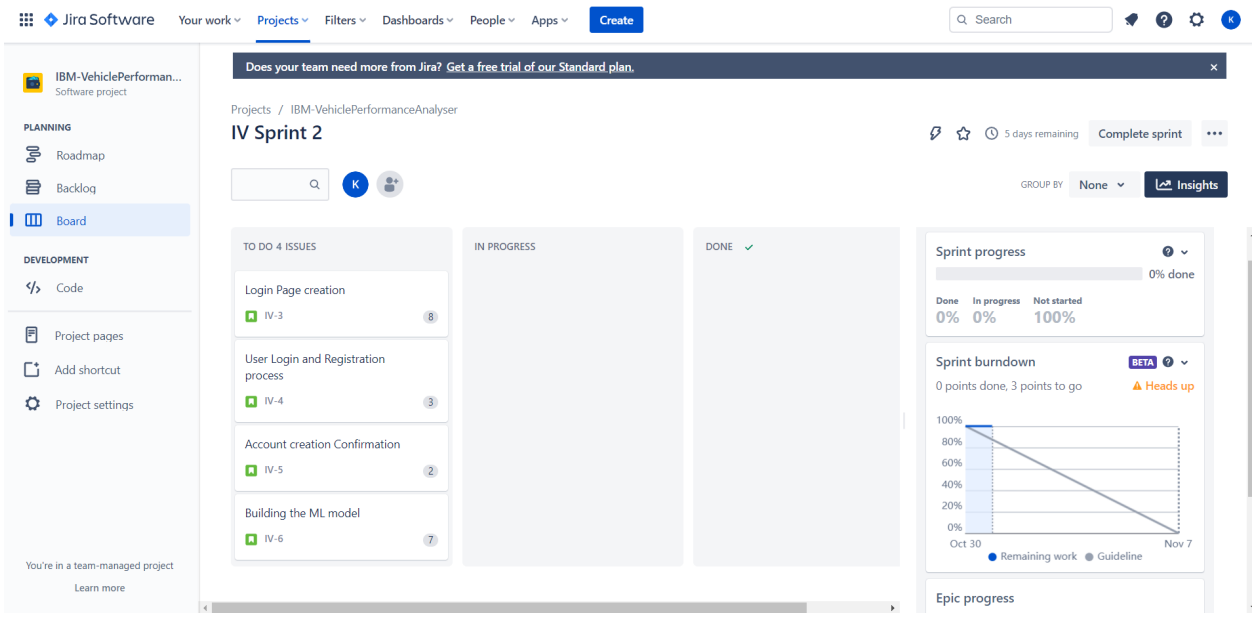
Sprint	**Total Story Points**	**Duration**	**Sprint Start Date**	**Sprint End Date (Planned)**	**Story Points Completed (as on Planned End Date)**	**Sprint Release Date (Actual)**
Sprint-1	15	6 Days	24 Oct 2022	29 Oct 2022	15	30 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	04 Nov 2022
Sprint-3	15	6 Days	07 Nov 2022	12 Nov 2022	15	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 Reports from JIRA


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IBM-VehiclePerforman...
Software project
PLANNING
Roadmap
Backlog
Board
Reports
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / IBM-VehiclePerformanceAnalyser
IV Sprint 3
0 days remaining
Complete sprint
GROUP BY
None
Insights
TO DO
IN PROGRESS
DONE 2 ISSUES
Vehicle data collection
IV-8
5
Query Processing
IV-9
10


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IBM-VehiclePerforman...
Software project
PLANNING
Roadmap
Backlog
Board
Reports
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / IBM-VehiclePerformanceAnalyser
IV Sprint 4
0 days remaining
Complete sprint
GROUP BY
None
Insights
TO DO
IN PROGRESS
DONE 1 ISSUE
Report generation
IV-10
20


vehicle-perf-analyser.atlassian.net/jira/software/projects/IV/boards/1/backlog
Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IBM-VehiclePerforman...
Software project
PLANNING
Roadmap
Backlog
Board
Reports
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / IBM-VehiclePerformanceAnalyser
Backlog
Epic
Insights
IV Sprint 2 30 Oct – 7 Nov (4 issues)
Complete sprint
IV-3 Login Page creation
IV-4 User Login and Registration process
IV-5 Account creation Confirmation
IV-6 Building the ML model
DONE
Create issue
Backlog (0 issues)
Create sprint
Your backlog is empty.
Create issue
Type here to search
26°C Mostly clear
ENG
00:00
19-11-2022


Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
IBM-VehiclePerforman...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / IBM-VehiclePerformanceAnalyser
IV Sprint 2
5 days remaining
Complete sprint
GROUP BY
None
Insights
TO DO 4 ISSUES
IN PROGRESS
DONE
Login Page creation
IV-3
User Login and Registration process
IV-4
Account creation Confirmation
IV-5
Building the ML model
IV-6
Sprint progress
0% done
Done
In progress
Not started
0%
0%
100%
Sprint burndown
BETA
0 points done, 3 points to go
Heads up
100%
80%
60%
40%
20%
0%
Oct 30
Nov 7
Remaining work
Guideline
Epic progress

7. CODING & SOLUTIONING

7.1 Feature 1 - Register/Login

Get the username and password from the user and use it to create an account on the application. Then allow the user to login to the application using the same credentials created.

7.2 Feature 2 - Predict Performance

Get input from the user about the various metrics related to the vehicle and make predictions about the performance based on the mileage of the car.

7.3 Feature 3 - Predict Similar Cars

Get input from the user of the various metrics of the car and use the metrics to predict similar cars to the car that had its metrics entered by the user of the web application.

[The appendix contains all of the source code for all the features implemented in the application]

8. TESTING

8.1 Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	BUG ID
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to see the Login/Signup in the homepage when user opens the website		1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup displayed or not		Login/Signup page should display	Working as expected	Pass		
LoginPage_TC_OO2	UI	Home Page	Verify the UI elements in Login/Signup popup		1.Enter URL and click go 2.Verify login/Singup appears with below UI elements: a.Username text box b.password text box c.Login button d.Sign Up button		Application should show below UI elements: a.email text box b.password text box c.Login button d.Sign Up button in different colour	Working as expected	Pass		
LoginPage_TC_OO3	Functional	Home page	Verify user is able to sign up to the application		1.Enter URL and click go 2.Enter a username in name text box 3.Enter a password in	User name: naveen password:	The application should accept the user if the user does not already exitst	Working as expected	Pass		

					password text box 4.Click on Sign Up button	pass word					
LoginPage_ TC_OO3	Functi onal	Home page	Verify user is able to log into applicati on with Valid credentia ls		1.Enter URL and click go 2.Enter Valid username in name text box 3.Enter valid password in password text box 4.Click on login button	User nam e: nave en pass word : pass word	User should navigate to user account homepage	Working as expected	Pass		
LoginPage_ TC_OO4	Functi onal	Login page	Verify user is able to log into applicati on with InValid credentia ls		1.Enter URL and click go 2.Enter inValid username in name text box 3.Enter invalid password in password text box 4.Click on login button	User nam e: fivhif vcjkd c pass word : fiuvn sdkfj veuf	Application should show 'Incorrect username or password ' validation message.	Working as expected	Pass		
LoginPage_ TC_OO5	Functi onal	Login page	Verify user is able to sign up to applicati on with already existing credentia		1.Enter URL and click go 2.Verify login/Singup appears with below UI elements: a.Username text box b.password	User nam e: nave en pass word : pass wd_	Application should show that an account with the name already exists	User overwritt en	Fail	Check for already existing user made and the bug is fixed	BUG-1

			ls		text box c.Login button d.Sign Up button	Test					
PredictPage_TC_OO1	UI	Input page	Verify the UI elements for the required input appears on the page		1.Enter URL and click go 2.Enter a valid username and password and click on Login 3. Verify if the following UI elements appear: a. Title of the app b. Kilometers Driven c. Fuel Type d. Transmission Type e. Engine Capacity f. Engine Power g. Number of Seats h. Car Price i. Get Mileage Button (in blue)		Application should display the required UI elements properly	Working as expected	Pass		

PredictPage_TC_OO2	Functional	Input page	Verify that the user is able to see the input boxes and the input button for inputting data		1.Enter URL and click go 2.Enter a valid username and password and click on Login 3. Verify the Vehicle Performance Analyzer page appears or not		Application should display the page allowing users to input data to the app	Working as expected	Pass		
PredictPage_TC_OO3	Functional	Input page	Verify that the user is able to input data to the application		1.Enter URL and click go 2.Enter a valid username and password and click on Login 3. Enter the details required to make prediction 4. Verify that the input is taken by the application and the prediction made is returned to the user		Application should take the input and redirect the user to the outputs page	Unhandled exception	Fail	Exception handled by fixing the error in app.py	BUG-02

ResultsPage_TC_001	Functional	Results page	Verify that predicted mileage is shown		1. Login into the application with valid credentials 2. Enter the requested data in the input page and submit the form 3. The user is redirectedd to the results page	Kilometers Driven: 100 Fuel Type: Diesel Transmission Type: Manual Engine Capacity: 1200 Engine Power: 120 Number of Seats: 7 Car Price: 2000000	Applicaiton should display the expected mileage and Cars that give similar mileage	Working as expected	Pass		

ResultsPage_TC_002	UI	Results page	Responsive behaviour across different screen sizes		1. Execute Test case ResultsPag_TC_001 with any test data 2. Enter browser Developer tools 3. Check the page layout with different screen sizes and devices		The application must be able to resize the elements in the page in order to fit into any screen size without overflow of the page	Working as expected	Pass		

8.2 User Acceptance Testing

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

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	3	5	1	3	12
Duplicate	0	1	3	7	11
External	2	1	0	1	4
Fixed	3	7	4	17	31
Not Reproduced	0	5	1	0	6
Skipped	0	0	1	2	3
Won’t Fix	0	0	2	1	3
Totals	8	19	12	31	70

9. RESULTS

9.1 Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - 0.980, MSE - 3.257, RMSE - 1.805, R2 score - 0.836	Test Data Report: Metrics for regression: mean absolute error: 0.980 mean squared error: 3.257 root mean squared error: 1.805 r2 score: 0.836
2.	Tune the Model	Validation Method - R2 Score	

10. ADVANTAGES & DISADVANTAGES

The application can aid the user in analyzing the performance of the vehicle they are using based on the details of the vehicle. The model has been trained on a dataset which was acquired recently so that it allows a higher level of accuracy for the predictions. Similar vehicles to the vehicle input are also made available to the user. The disadvantages range from the fact that the performance of the vehicle is made based on the mileage alone. Also, the application works only with cars and not other vehicles.

11. CONCLUSION

It is a significant and fascinating challenge to be able to gauge a car's performance level. Our main objective was to forecast vehicle performance in order to enhance specific vehicle behavior. performance assessment of the vehicle taking into account factors like horsepower, cylinder count, fuel type, and engine type. The health of the car is predicted based on variables including horsepower, cylinder count, fuel type, and engine type. In order to maximize the performance effectiveness of the vehicle, we studied the components using a variety of well-known machine learning techniques, including linear regression, decision trees, and random forests. The "hot topics" in automotive engineering right now are the power, longevity, and range of automobile traction batteries. In this instance, we also take mileage performance into account.

12. FUTURE SCOPE

In the future, the project can be expanded by including more parameters to analyze the performance of the vehicles and also to expand the project to be able to predict the performance of other vehicles like motorbikes, trucks, etc. More models can also be experimented with to find the model which gives better performance for predicting the performance of vehicles.

13. APPENDIX

Source Code

app.py

```
import numpy as np

import csv

import sys

import requests



from flask import Flask, url_for, render_template, request, redirect,
session

from flask_sqlalchemy import SQLAlchemy

from dotenv import dotenv_values



config = dotenv_values(".env")

app = Flask(__name__)

app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///users.db'

db = SQLAlchemy(app)



API_KEY = config['API_KEY']



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}




class User(db.Model):

    id = db.Column(db.Integer, primary_key=True)

    username = db.Column(db.String(100), unique=True)

    password = db.Column(db.String(100))


    def __init__(self, username, password):

        self.username = username

        self.password = password




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

def index():

    if session.get('logged_in'):

        return render_template('home.html')

    else:

        return render_template('index.html', message="Welcome!")




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

def register():

    if request.method == 'POST':
```

```
        try:

            db.session.add(

                User(username=request.form['username'],
password=request.form['password']))

            db.session.commit()

            return redirect(url_for('login'))

        except:

            return render_template('index.html', message="User Already
Exists")

    else:

        return render_template('register.html')




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

def results():

    if request.method == 'POST':

        kms_driven = request.form['Kilometers_Driven']

        fuel_type = request.form['Fuel_Type']

        transmission = request.form['Transmission']

        engine_cc = request.form['Engine_CC']

        power = request.form['Power']

        seats = request.form['Seats']

        carprice = request.form['Price']

        t1 = [[int(kms_driven), str(fuel_type), str(transmission), int(

            engine_cc), int(power), int(seats), int(carprice), ]]

        # t1 = [[1, "Maruti Swift VDI", "Maruti", "Delhi", 2014,
int(kms_driven), str(fuel_type), str(transmission), "First", int(

        # engine_cc), int(power), int(seats), int(carprice), ]]

        payload_scoring = {"input_data": [{"field": [
```

```
            ["Kilometers_Driven", "Fuel_Type", "Transmission", "Engine
CC", "Power", "Seats", "Price"]], "values": t1}]}

        response_scoring =
requests.post('https://eu-de.ml.cloud.ibm.com/ml/v4/deployments/acb7733e-5
447-4b2d-b358-312456c41d1f/predictions?version=2022-11-17',
json=payload_scoring,

                                        headers={'Authorization': 'Bearer
' + mltoken})

        print("Scoring response")

        prediction =
response_scoring.json()["predictions"][0]["values"][0][0]

        print(prediction)

        final_prediction = round(prediction, 2)



        suggested_cars = []

        csv_file = csv.reader(open('CARS_2.csv', "r"), delimiter=",")

        for row in csv_file:

            if(row[12] == 'Mileage Km/L'):

                continue

            else:

                num1 = float(final_prediction)

                num2 = float(row[12])

                if abs(num1-num2) <= 1:

                    suggested_cars.append(row[1])

                    # print(row[1])

        suggested_cars = set(suggested_cars)



        return render_template('results.html', z=str(final_prediction),
car_list = suggested_cars)

    #     try:
```

```
    #         db.session.add(User(username=request.form['username'],
password=request.form['password']))

    #         db.session.commit()

    #         return redirect(url_for('login'))

    #     except:

    #         return render_template('index.html', message="User Already
Exists")

    # else:

    #     return render_template('register.html')




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

def login():

    if request.method == 'GET':

        return render_template('login.html')

    else:

        u = request.form['username']

        p = request.form['password']

        data = User.query.filter_by(username=u, password=p).first()

        if data is not None:

            session['logged_in'] = True

            return redirect(url_for('index'))

        return render_template('index.html', message="Incorrect Details")



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

def logout():

    session['logged_in'] = False

    return redirect(url_for('index'))
```

```
if(__name__ == '__main__'):
    app.secret_key = "ThisIsNotASecret:p"
    with app.app_context():
        db.create_all()
    app.run(debug=config['DEBUG'])
```

suggestion.py

```
import csv
import sys


# input number you want to search
number = input('Enter number to find\n')


# read csv, and split on "," the line
csv_file = csv.reader(open('CARS_2.csv', "r"), delimiter=",")




# loop through the csv list
for row in csv_file:
    # if current rows 2nd value is equal to input, print that row
    if(row[12] == 'Mileage Km/L'):
        continue
    else:
        num1 = float(number)
        num2 = float(row[12])
```

```
        if abs(num1-num2) <= 1:

            print(row[1])
```

base.html

```
<!DOCTYPE html>

<html lang="en">

<head>

    <title>Vehicle Performance Analyser</title>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link rel="icon" type="image/png" href="{{ url_for('static',
filename='images/icons/favicon.ico') }}"/>

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.
css" rel="stylesheet"
integrity="sha384-Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy
9Bv1WTRi" crossorigin="anonymous">



</head>

<body>

    <div class="w-auto min-vh-100 d-flex">

        {% block content %}{% endblock %}

    </div>

    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle
.min.js"
integrity="sha384-OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f
6V8Qbsw3" crossorigin="anonymous"></script>

    <script src="{{ url_for('static', filename='js/main.js')}}"></script>

</body>

</html>
```

home.html

```
{% extends 'base.html' %}

{% block content %}

<div class="d-flex h-auto w-100 justify-content-center align-items-center
flex-grow flex-column">

    <h1 class="mb-5">Vehicle Performance Analyzer</h1>

    <div class="container w-md-75">

        <form class="row g-3 border" action="/results" method="POST">

            <div class="col-md-6">

                <label for="Kilometers_Driven"
class="form-label">Kilometers Driven</label>

                <input class="form-control" type="number"
name="Kilometers_Driven" placeholder="Eg: 15000" min="0" required>

            </div>

            <div class="col-md-6">

                <label for="Fuel_Type" class="form-label">Fuel
Type</label>

                <select class="form-select" name="Fuel_Type"
placeholder="Fuel Type" min="0" required>

                    <option value="-1" selected>SELECT</option>

                    <option value="0">CNG</option>

                    <option value="1">Diesel</option>

                    <option value="2">Petrol</option>

                    <option value="3">LPG</option>

                    <option value="4">Electric</option>

                </select>

            </div>
```

```
            <div class="col-md-6">

                <label for="Fuel_Type" class="form-label">Transmission
Type</label>

                <select class="form-select" name="Transmission"
placeholder="Transmission" min="0" required>

                    <option value="-1" selected>SELECT</option>

                    <option value="0">Manual</option>

                    <option value="1">Automatic</option>

                </select>

            </div>

            <div class="col-md-6">

                <label for="Engine_CC" class="form-label">Engine
Capacity</label>

                <input class="form-control" type="number" name="Engine_CC"
placeholder="Eg: 1000" min="0" required>

            </div>

            <div class="col-md-4">

                <label for="Power" class="form-label">Engine Power</label>

                <input class="form-control" type="number" name="Power"
placeholder="Eg: 80" min="0" required>

            </div>

            <div class="col-md-4">

                <label for="Seats" class="form-label">Number of
Seats</label>

                <input class="form-control" type="number" name="Seats"
placeholder="Eg: 5" min="0" required>

            </div>

            <div class="col-md-4">

                    <label class="form-label" for="Price">Car
Price</label>

                    <input class="form-control" type="number" name="Price"
placeholder="Eg: 500000" min="0" required>
```

```
            </div>

            <div class="col-12">

                <button type="submit" class="btn btn-primary w-100
text-center">Get Mileage</button>

            </div>

        </form>

    </div>

</div>




    <!-- <button class="login100-form-btn">

            <a class="login100-form-btn" href="/logout">Logout</a>

        </button> -->

    {% endblock %}
```

index.html

```
{% extends 'base.html' %}

{% block content %}

    <div class="d-flex h-auto w-100 justify-content-center
align-items-center flex-grow">

        <div class="container text-center">

                <h1 class="title-1 mb-3">{{ message }}</h1>

                <a class="btn btn-md btn-primary me-3" href="{{
url_for('login')}}">Login</a>

                <a class="btn btn-md btn-secondary" href="{{
url_for('register')}}">Register</a>

            </div>
```

```
        </div>

    </div>

{% endblock %}
```

login.html

```
{% extends 'base.html' %}

{% block content %}

<div class="d-flex h-auto w-100 justify-content-center align-items-center
flex-grow flex-column">

    <h1 class="mb-3">Login</h1>

    <div class="container-flex border p-3">

        <form  method="POST" action="/login">

            <div class="mb-3">

                <label for="username" class="form-label">Username</label>

                <input type="text" class="form-control" id="username"
name="username" placeholder="Your Name"  required>

            </div>

            <div class="mb-3">

                <label for="password" class="form-label">Password</label>

                <input type="password" class="form-control"
name="password" placeholder="Password"  required>

            </div>

            <button type="submit" class="btn btn-primary">Login</button>

            <a href="{{ url_for('register')}}" class="btn
btn-outline-secondary">Sign Up</a>

        </form>

    </div>
```

```
</div>

{% endblock %}
```

register.html

```
{% extends 'base.html' %}

{% block content %}

<div class="d-flex h-auto w-100 justify-content-center align-items-center
flex-grow flex-column">

    <h1 class="mb-3">Register</h1>

    <div class="container-flex border p-3">

        <form  method="POST" action="/register">

            <div class="mb-3">

                <label for="username" class="form-label">Username</label>

                <input type="text" class="form-control" id="username"
name="username" placeholder="Your Name"  required>

            </div>

            <div class="mb-3">

                <label for="password" class="form-label">Password</label>

                <input type="password" class="form-control"
name="password" placeholder="Password"  required>

            </div>

            <button type="submit" class="btn btn-primary">Sign Up</button>

            <a href="{{ url_for('login')}}" class="btn
btn-outline-secondary">Login</a>

        </form>

    </div>

</div>
```

```
    <!-- <div class="limiter">

        <div class="container-login100">

            <div class="wrap-login100 p-t-50 p-b-90">

                <form class="login100-form validate-form flex-sb flex-w"
method="POST" action="/register">

                    <span class="login100-form-title p-b-51">

                        Register

                    </span>

                    <div class="wrap-input100 validate-input m-b-16"
data-validate = "Username is required">

                        <input class="input100" type="text"
name="username" placeholder="Username">

                        <span class="focus-input100"></span>

                    </div>

                    <div class="wrap-input100 validate-input m-b-16"
data-validate = "Password is required">

                        <input class="input100" type="password"
name="password" placeholder="Password">

                        <span class="focus-input100"></span>

                    </div>

                    <div class="flex-sb-m w-full p-t-3 p-b-24">

                        <div>

                            <a href="{{url_for('login')}}" class="txt1">

                                Log In

                            </a>

                        </div>

                    </div>

                    <div class="container-login100-form-btn m-t-17">

                        <button class="login100-form-btn">

                            Register
```

```
                        </button>

                    </div>

                </form>

            </div>

        </div>

    </div> -->

{% endblock %}
```

results.html

```
{% extends 'base.html' %}

{% block content %}

    <div class="d-flex h-auto w-100 justify-content-center
align-items-center flex-grow">

        <div class="container text-center">

                <div class="container text-center">

                <h1 class="title-1 mb-3">Predicted Mileage</h1>

                <h4>{{z}} Km/L</h4>

                </div>

                <br>

                <br>

                <!-- <div style = "overflow:scroll"> -->

                <div>

                <h1 class="title-1 mb-3">Cars suggested based on users'
input demands</h1>

                <ul style= "position: relative; overflow: scroll; top:
20px;bottom: 20px; left: 20px; right: 20px; border: 2px solid #ccc;
font-size: 16px; font-family: Arial, sans-serif;
-webkit-overflow-scrolling: touch">
```

```
                {% for car in car_list %}

                <option style= "padding: 10px 20px; border-bottom: 1px
solid #ccc" value= "{{car}}">{{car}}</option>

                {% endfor %}

                </ul>

                </div>

                <!-- </div> -->

            <!-- </div> -->

        </div>

    </div>

    </div>

{% endblock %}
```

main.js

```
(function ($) {

    "use strict";

    /*==================================================================

    [ Validate ]*/

    var input = $('.validate-input .form-control');

    $('.validate-form').on('submit',function(){

        var check = true;


        for(var i=0; i<input.length; i++) {

            if(validate(input[i]) == false){

                showValidate(input[i]);

                check=false;
```

```
            }

        }


        return check;

    });


    $('.validate-form .input100').each(function(){

        $(this).focus(function(){

           hideValidate(this);

        });

    });

    function validate (input) {

        if($(input).attr('type') == 'email' || $(input).attr('name') ==
'email') {


if($(input).val().trim().match(/^([a-zA-Z0-9_\-\.]+)@((\[[0-9]{1,3}\.[0-9]
{1,3}\.[0-9]{1,3}\.)|(([a-zA-Z0-9\-]+\.)+))([a-zA-Z]{1,5}|[0-9]{1,3})(\]?)
$/) == null) {

                return false;

            }

        }

        else {

            if($(input).val().trim() == ''){

                return false;

            }

        }

    }


    function showValidate(input) {
```

```
        var thisAlert = $(input).parent();


        $(thisAlert).addClass('alert-validate');

    }

    function hideValidate(input) {

        var thisAlert = $(input).parent();


        $(thisAlert).removeClass('alert-validate');

    }

})(jQuery);
```

GitHub Link & Demo Link

Project GitHub link: https://github.com/IBM-EPBL/IBM-Project-5280-1658755490

Project Demo link: https://youtu.be/bDleKW7gJRg