Date	14 November 2022
Team ID	PNT2022TMID21127
Project Name	Car Resale Value Prediction

CAR RESALE VALUE PREDICTION

BY

NARENDRAN G

NIKHIL MADHAV M

SUDHARSAN K

VALLARASU R

1.INTRODUCTION

In this project we have used different algorithms with different techniques for developing Car resale value prediction systems considering different features of the car. In a nutshell, car resale value prediction helps the user to predict the resale value of the car depending upon various features like kilo meters driven, fuel type, etc.
This resale value prediction system is made for general purpose to just predict the amount that can be roughly acquired by the user. We try to predict the amount of resale by best 70% accuracy so the user can get estimated value before he resales the car and doesn't make a deal in loss.

1.1 PROJECT OVERVIEW

With difficult economic conditions, it is likely that sales of second-hand imported (reconditioned) cars and used cars will increase. In many developed countries, it is common to lease a car rather than buying it outright. After the lease period is over, the buyer has the possibility to buy the car at its residual value, i.e. its expected resale value. Thus, it is of commercial interest to sellers/financers to be able to predict the salvage value (residual value) of cars with accuracy.
In order to predict the resale value of the car, we proposed an intelligent, flexible, and effective system that is based on using regression algorithms. Considering the main factors which would affect the resale value of a vehicle a regression model is to be built that would give the nearest resale value of the vehicle. We will be using various regression algorithms and algorithm with the best accuracy will be taken as a solution, then it will be integrated to the web-based application where the user is notified with the status of his product.

1.2 PURPOSE

The main idea of making a car resale value prediction system is to get hands-on practice for python using Data Science. Car resale value prediction is the system to predict the amount of resale value based on the parameters provided by the user. User enters the details of the car into the form given and accordingly the car resale value is predicted.
The system is defined in the python language that predicts the amount of resale value based on the given information. The system works on the trained dataset of the machine learning program that evaluates the precise value of the car. User can enter details only of fields like purchase price of car, kilo meters driven, fuel of car, year of purchase.
Car resale value prediction system is made with the purpose of predicting the correct valuation of used cars that helps users to sell the car remotely with perfect valuation and without human intervention in the process to eliminate biased valuation.

2. LITERATURE SURVEY

1. **Car Price Prediction Using Machine Learning (2019) - Enis gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric.**

   In this work, several distinct attributes are analyzed for the reliable and accurate prediction. The work is to build a model to predict the resale price of cars in Bosnia and Herzegovina

2. Prediction of Used Car Price Based on Supervised Learning Algorithm (2021) - Feng Wang, Xusong Zhang, Qiang Wang

In this work, Extra Trees Regressor, Random Forest Regressor was used. Finally, the algorithm was optimized by using the hyperparameter function. The results show that R2 = 0.9807 obtained from extreme random numbers is the best performance. The algorithm was obtained and validated with new data to derive the final algorithm model.

3. Car resale value prediction using regression method

This paper study statistical models for forecasting the resale prices of used cars. An empirical study Is performed to explore the contributions of different degrees of freedom in the modelling process to the forecast accuracy. First, a comparative analysis of alternative prediction methods provides Evidence that random forest regression is particularly effective for resale price forecasting. Second, The empirical results demonstrate the presence of heterogeneity in resale price forecasting and Identify methods that can automatically overcome its detrimental effect on the forecast accuracy. Finally, the study confirms that the sellers of used cars possess informational advantages over. Market research agencies, which enable them to forecast resale prices more accurately. This Implies that sellers have an incentive to invest in in-house forecasting solutions, instead of basing. Their pricing decisions on externally generated residual value estimates.

4. **Marcus Collard "Price Prediction for Used Cars": Mid Sweden University. June 8, 2022**

Cars of a particular make, model, year, and set of features start out with a price set by the manufacturer. As they age and are to their unique history. The more this sets them apart from comparable cars, the harder they become resold as used, they are subject to supply-and-demand pricing for their particular set of features, in addition to evaluate with traditional methods.

5. **Abdulla AlShared "Used Cars Price Prediction and Valuation using Data Mining Techniques": Rochester Institute of Technology RIT Dubai DEC 2021**

Using data mining and machine learning approaches, this project proposed a scalable framework for Dubai based used cars price prediction. Buyanycar.com website

was scraped using the Parse Hub scraping tool to collect the benchmark data. An efficient machine learning model is built by training, testing, and evaluating three machine learning regressors named Random Forest Regressor, Linear Regression, and Bagging Regressor.

2.1 EXISTING PROBLEM

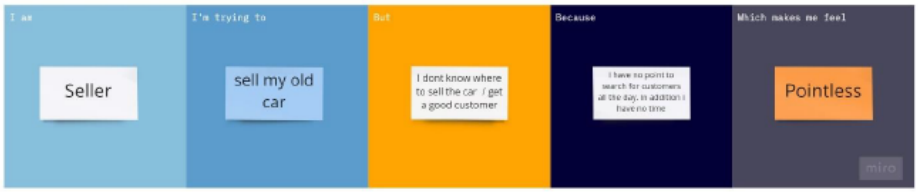
The prices of new cars in the industry is fixed by the manufacturer with some additional costs incurred by the Government in the form of taxes. So, customers buying a new car can be assured of the money they invest to be worthy. But due to the increased price of new cars and the incapability of customers to buy new cars due to the lack of funds, used cars sales are on a global increase. There is a need for a used car price prediction system to effectively determine the worthiness of the car using a variety of features. Even though there are websites that offers this service, their prediction method may not be the best. Besides, different models and systems may contribute on predicting power for a used car's actual market value. It is important to know their actual market value while both buying and selling.

2.2 REFERENCES –

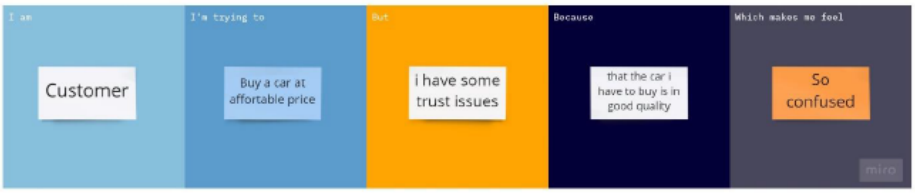
- https://www.irjet.net/archives/V8/i5/IRJET-V8I5490.pdf
- http://cs229.stanford.edu/proj2019aut/data/assignment_308832_raw/26612934.pdf
- https://www.enjoyalgorithms.com/blog/car-resale-value-predictor-using-random-forest-regressor
- https://towardsdatascience.com/predicting-used-car-prices-with-machine-learning-techniques-8a9d8313952

2.3 PROBLEM STATEMENT DEFINITION

Problem Statement for Seller:



Problem Statement for Customer:

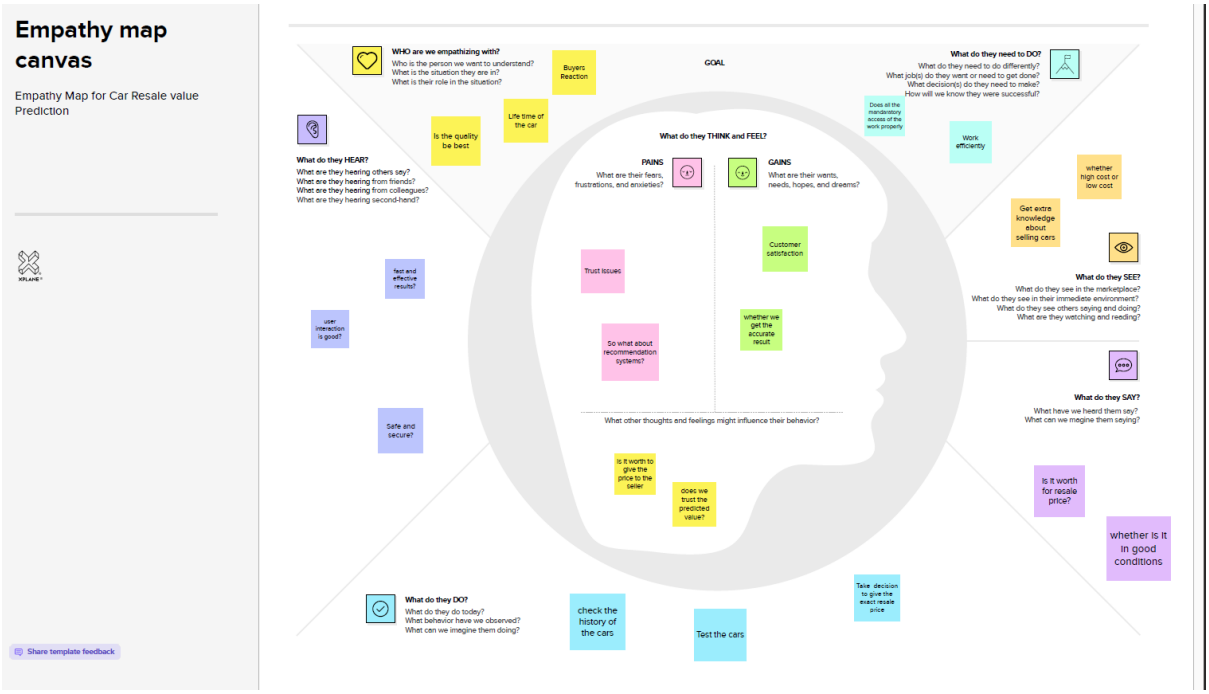


Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Student	Buy a car to travel at least 30 mins a day	i can't able to find a car for my requirements	I have no place to search for a second hand car	So upset
PS-2	Driver	Buy a car for my regular work purpose	I have lots of money to buy it and it has to be in good condition	there's a lots of variety of cars which is not suitable for cab purposes	Feel bad

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. The empathy map was originally created by Dave Gray and has gained much popularity within the agile community.



3.2 IDEATION & BRAINSTORMING

A mind map is a tool for the brain that captures the thinking that goes on inside your head. Mind mapping helps you think, collect knowledge, remember and create ideas. Most likely it will make you a better thinker.

3.3 PROPOSED SOLUTION

S.no.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To predict the resale value for Cars and provide the Customer with accurate and satisfactory results regarding the same.
2.	Idea / Solution description	To develop a Machine Learning Algorithm which predicts the resale value by using Python.
3.	Novelty / Uniqueness	The resale value of the Car can be predicted with a much higher and apt accuracy as the Customer expects.
4.	Social Impact / Customer Satisfaction	Customer Satisfaction plays a vital role in our project. If the customer is satisfied, then automatically the project will become popular & useful, thus leading to a great Social Impact.
5.	Business Model (Revenue Model)	A Revenue Model is a blueprint which shows how much revenue or gross income that will be earned through the sales and how those costs will be able to cover the operating costs and expenses.
6.	Scalability of the Solution	The Software is being deployed in Cloud, so it can be accessed by anyone who owns it from anywhere. It can even be accessed through mobile phones by customers. This project will provide the most accurate results by using algorithms related to the Customer and their preferences.

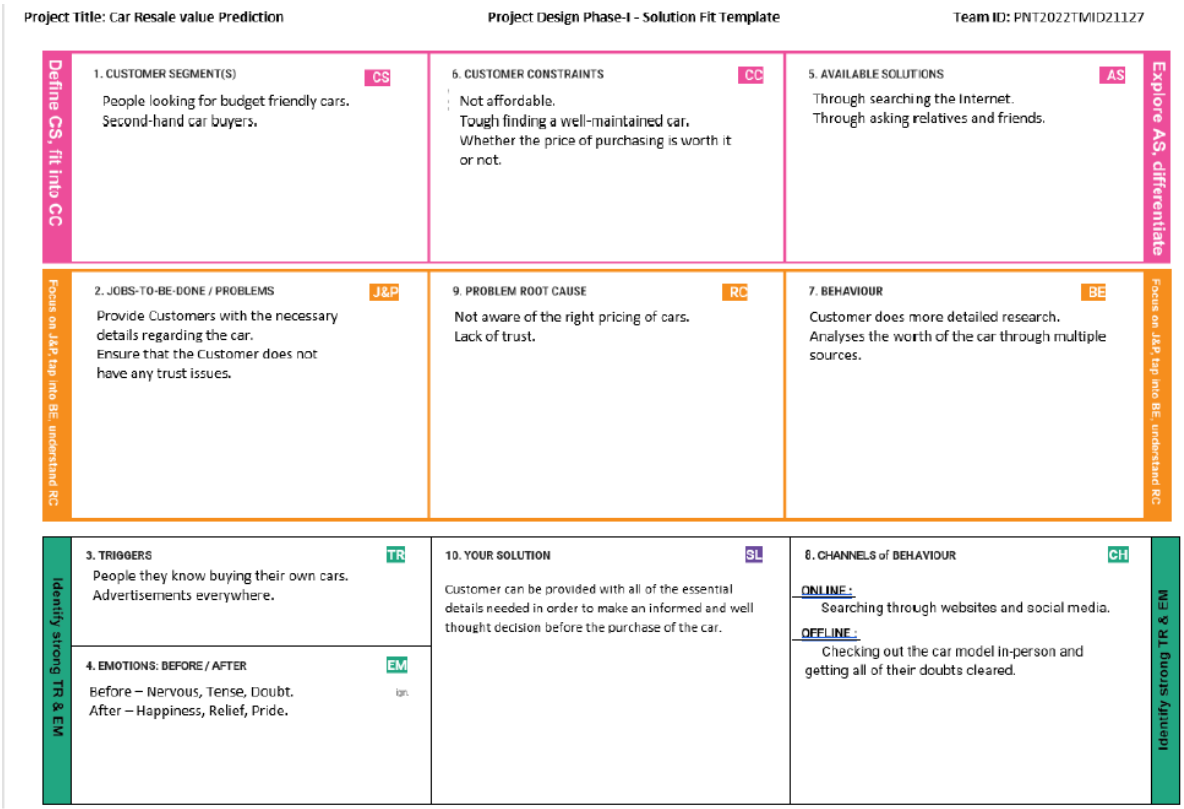
3.4 PROBLEM SOLUTION FIT

Problem – Solution Fit Template:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why

Purpose:

- Solve complex problems in a way that fits the state of your customers.
- Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behaviour.
- Sharpen your communication and marketing strategy with the right triggers and messaging.
- Increase touch-points with your company by finding the right problem-behaviour fit and building trust by solving frequent annoyances, or urgent or costly problems.
- Understand the existing situation in order to improve it for your target group.



4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS –

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration directly on website Registration through Gmail Registration through Google
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Dashboard	User details Used and new cars details Buy and sell cars Latest updates in automobile industry Customer reviews
FR-4	Car Registration	The user can add the details like vehicle price, model name and number, vehicle model, vehicle brand and mileage
FR-5	Value Prediction	Predicting the car resale value
FR-6	Car Ordering	The user can order the cars by past and top predictions as well as reseller's history and background mentioned

4.2 NON-FUNCTIONAL REQUIREMENTS –

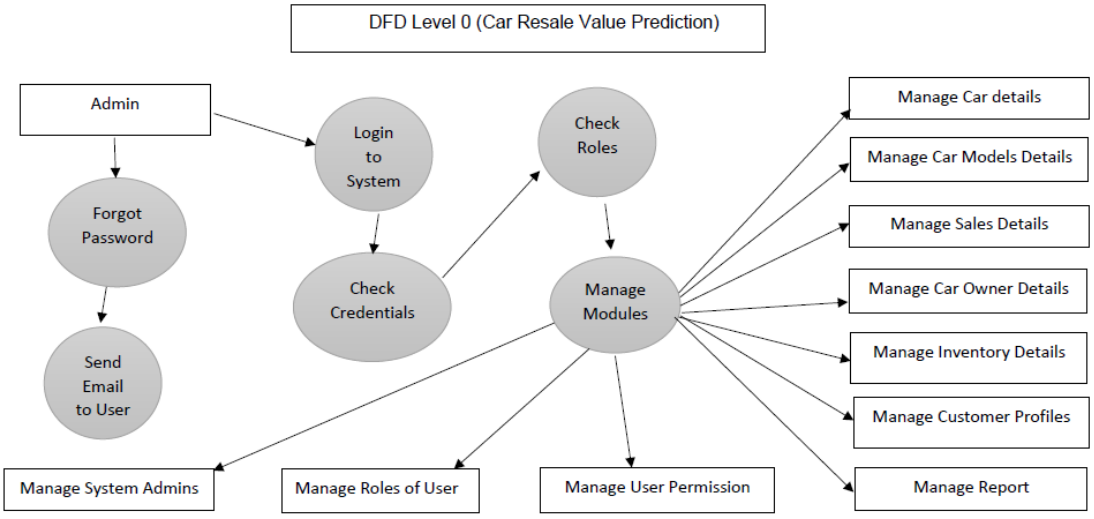
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Smart and user-friendly interface with proper design that can be provide a better experience for the user.
NFR-2	Security	Make sure that the all user information should be safe and protected.
NFR-3	Reliability	Provides high reliability by predicting values for cars. The ML model, which is responsible for prediction,

		should be accurate enough to predict prices and the error rate should be as low as possible.
NFR-4	Performance	Provides high performance by using some machine learning algorithms.
NFR-5	Availability	The website should be available to users 24x7. Any issues or errors will be addressed within the next 24 hours.
NFR-6	Scalability	Providing high scalability in predicting values for the cars.

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 ARCHITECTURE

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

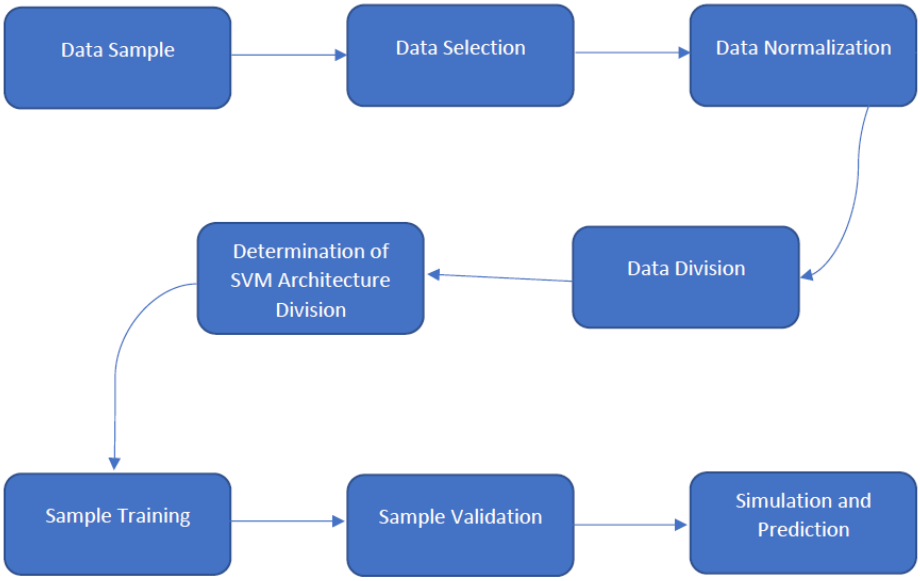
• Find the best tech solution to solve existing business problems.

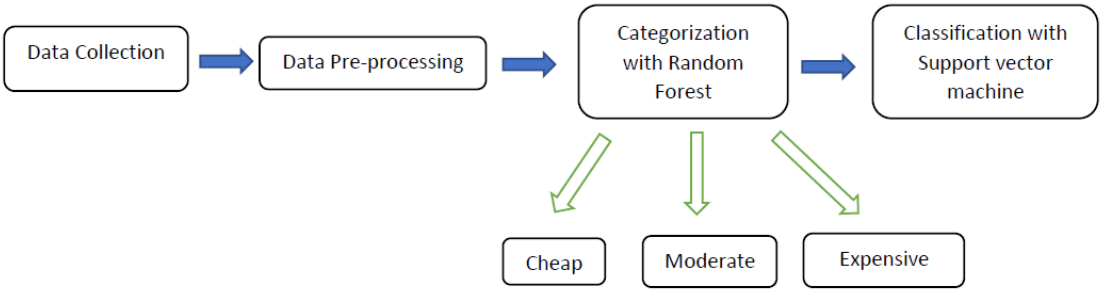
• Describe the structure, characteristics, behaviour, and other aspects of the software to project stakeholders.

• Define features, development phases, and solution requirements.

• Provide specifications according to which the solution is defined, managed, and delivered.

Solution Architecture Diagram:





TECHNICAL ARCHITECTURE

Technical Architecture (TA) is a form of IT architecture that is used to design computer systems. It involves the development of a technical blueprint with regard to the arrangement, interaction, and interdependence of all elements so that system-relevant requirements are met.

Technical Architecture:

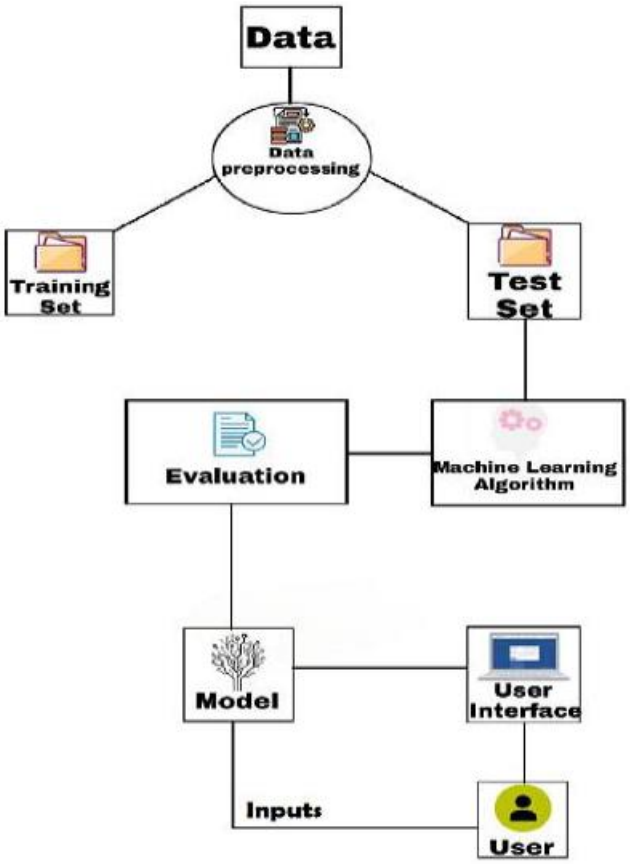


Table-1 : Components & Technologies:

S.no	Component	Description	Technology
1.	User Interface	User interacts with the application using Web UI	HTML, CSS, JavaScript
2.	Application Logic-1	The user create login id and password	Python
3.	Application Logic-2	The user add the car's details in the application	Python
4.	Application Logic-3	The user can place their orders by best predicted values	Python
5.	Database	Stores the user's id and vehicles information in database	MySQL / SQL Server
6.	Cloud Database	The dataset stored in IBM cloud	IBM cloud
7.	File Storage	Stores the users and vehicles details	Local Storage System
8.	Machine Learning Model	The different types of ML models are used for the application. Like, KNN, Random Forest Regression	Regression Algorithm
9.	Infrastructure (Server / Cloud)	Application Deployment : Local System / Cloud Local Server Configuration : User's local data stored in local network Cloud Server Configuration : Car details, prediction process and car value are in cloud	Local

Table-2: Application Characteristics:

S.no	Characteristics	Description	Technology
1.	Open-Source Frameworks	It is a code, that is designed to be accessible publicly	HTML, CSS, Python Flask
2.	Security Implementations	Protection for authentication process and securing data in cloud	Encryptions
3.	Scalable Architecture	It is the property of a model to handle a raising amount of work by adding resources to the model. It consists of 3 tiers	Web Server - HTML, CSS, Java Script Database Server - IBM Cloud Application Server - Python Flask
4.	Availability	It is for use of distributed servers and load balancers	IBM cloud
5.	Performance	It is defines the function of application and how responsive to the user	IBM Cloud environment

5.3 USER STORIES

In software development and product management, a user story is an informal, natural language description of one or more features of a software system. A user story is a tool used in Agile software development to capture a description of a software feature from an end-user perspective. A user story describes the type of user, what they want and why. A user story helps to create a simplified description of a requirement.

User stories are often recorded on index cards, on Post-it notes, or in project management software. Depending on the project, user stories may be written by various stakeholders such as clients, users, managers or development team members.

User Stories: Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the car resale value prediction application by entering my email id, password and confirming my password	I can access my account and dashboard	High	Sprint-1
		USN-2	I will receive confirmation email once I have registered for the application	I can receive confirmation email and click confirm to verify my account	High	Sprint-1
		USN-3	I can register for the application through Google, LinkedIn	I can register and access the dashboard with Google and LinkedIn login	Medium	Sprint-2
	Login	USN-4	As a User, I can login to the application by entering the email id and password	I can login using email and password	High	Sprint-1
	Dashboard	USN-5	I can access the dashboard after login and view the details about different models of used and used cars	I can access the dashboard and predicted values of used cars	Low	Sprint-2
Customer (Web user)	Registration	USN-6	As a user, I can register for the car resale value prediction web application by entering my email id, password and confirming my password	I can access my account and dashboard	High	Sprint-1
		USN-7	As a user, I will receive confirmation email once I have registered for the web application	I can receive confirmation email and click confirm to verify my account	High	Sprint-1
		USN-8	As a user, I can register for the web application through Google, LinkedIn	I can register and access the dashboard with Google and LinkedIn login	Medium	Sprint-2
	Login	USN-9	As a user, I can login to the web application by entering the email id and password	I can login using email and password	High	Sprint-3
	Dashboard	USN-10	As a user, I can access the dashboard after login and view the details about different models of used and used cars	I can access the dashboard and predicted values of used cars	Low	Sprint-4
Customer Care Executive	Customer Support	USN-11	As a user, I can contact the customer care and raise a query to them	I can contact the customer care and chat with them	High	Sprint-3
Administrator	Data Maintenance	USN-12	As a user, my data is maintained by the admin	Admin can access and maintains the customers data	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

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	Pre-process the data	USN-1	Collect Dataset	1	Low	Vallarasu R
Sprint-1		USN-2	Import required libraries	1	Low	Vallarasu R
Sprint-1		USN-3	Read and Clean dataset	3	Medium	Vallarasu R
Sprint-1	Model Building	USN-1	Split data into independent and dependent variables	3	Medium	Narendran G
Sprint-2		USN-2	Apply using regression model	3	Medium	Nikhil Madhav M
Sprint-2	Application Building	USN-1	Build Python Flask Application and HTML page	3	Medium	Narendran G
Sprint-2		USN-2	Execute and fest	3	Medium	Narendran G
Sprint-2	Train the Model	USN-1	Train machine learning model	3	Medium	Sudharsan K
Sprint-3		USN-2	Integrate flask	3	Medium	Sudharsan K
Sprint-3	Registration	USN-1	User can register for the car resale value prediction application by entering my email id, password and confirming my password.	5	High	Nikhil Madhav M
Sprint-3		USN-2	User will receive confirmation email once I have registered for the application	5	High	Sudharsan K
Sprint-3		USN-3	User can register for the application through google	1	Low	Sudharsan K
Sprint-4		USN-4	User can register for the application through Gmail	3	Medium	Narendran G
Sprint-4	Login	USN-5	User can log into the application by entering email & password	5	High	Nikhil Madhav M
Sprint-4	Dashboard	USN-6	User can access the dashboard after login and view the details about different models of used and used cars	5	High	Vallarasu R

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	8	2 Days	01 Nov 2022	02 Nov 2022	8	02 Nov 2022
Sprint-2	12	4 Days	03 Nov 2022	06 Nov 2022	10	07 Nov 2022
Sprint-3	14	5 Days	07 Nov 2022	11 Nov 2022	11	11 Nov 2022
Sprint-4	13	7 Days	12 Nov 2022	18 Nov 2022	3	18 Nov 2022

7. CODING & SOLUTIONING

7.1 Splitting the dataset

In this activity, the dependent and independent variables are to be identified. The first column (Result) in the cleaned dataset is the dependent variable which is dependent on the remaining different factors. The independent columns are considered as x and the dependent column as y.

```
Y = labeled.iloc[:,0].values
X = labeled.iloc[:,1:].values
Y=Y.reshape(-1,1)

from sklearn.model_selection import cross_val_score, train_test_split
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.3, random_state=3)
```

7.2 Choose the Appropriate model

We will be initially considering the Random Forest Regressor model and fit the data.

```
from sklearn.ensemble import RandomForestRegressor
from sklearn.metrics import r2_score

regressor = RandomForestRegressor(n_estimators = 1000,max_depth=10,random_state=34)
regressor.fit(X_train,np.ravel(Y_train,order='C'))
```

7.3 Integrate the Flask with Scoring End Point

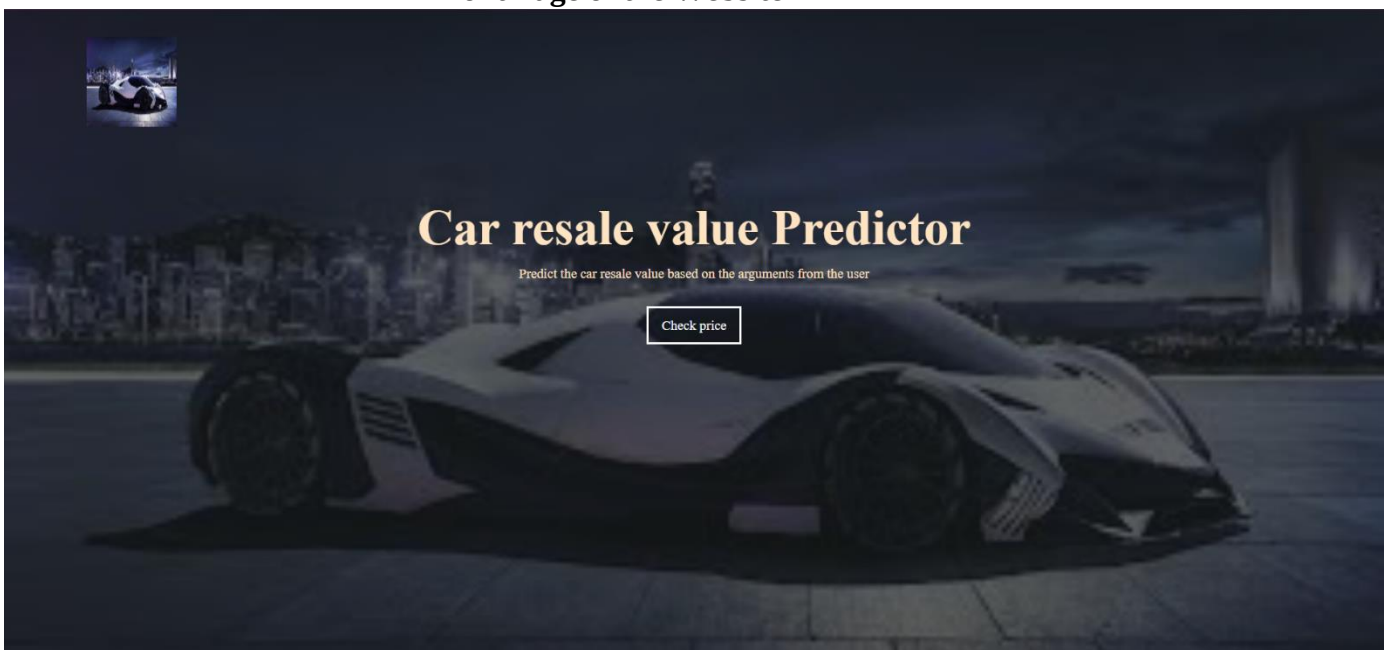
```
import pandas as pd
import numpy as np
from flask import Flask, render_template, Response, request
import pickle
from sklearn.preprocessing import LabelEncoder
import requests
```

```
API_KEY = "Uj-wf8uhJR3--c1cxhOPx3cWRIdP-mxNIRlkp7GPO_2g"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":A
PI_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(__name__)

def load_model(file='../Result/resale_model.sav'):
  return pickle.load(open(file, 'rb'))

@app.route('/')
def index():#main page
  return render_template('car.html')

@app.route('/predict_page')
def predict_page():#predicting page
  return render_template('value.html')

@app.route('/predict', methods=['GET','POST'])
def predict():
  reg_year = int(request.args.get('regyear'))
  powerps = float(request.args.get('powerps'))
  kms= float(request.args.get('kms'))
  reg_month = int(request.args.get('regmonth'))

  gearbox = request.args.get('geartype')
  damage = request.args.get('damage')
  model = request.args.get('model')
  brand = request.args.get('brand')
  fuel_type = request.args.get('fuelType')
  veh_type = request.args.get('vehicletype')

  new_row = {'yearOfReg':reg_year, 'powerPS':powerps, 'kilometer':kms,
       'monthOfRegistration':reg_month, 'gearbox':gearbox,
       'notRepairedDamage':damage,
       'model':model, 'brand':brand, 'fuelType':fuel_type,
       'vehicletype':veh_type}

  print(new_row)

  new_df = pd.DataFrame(columns=['vehicletype','yearOfReg','gearbox',
    'powerPS','model','kilometer','monthOfRegistration','fuelType',
    'brand','notRepairedDamage'])
  new_df = new_df.append(new_row, ignore_index=True)
  labels = ['gearbox','notRepairedDamage','model','brand','fuelType','vehicletype']
  mapper = {}
```

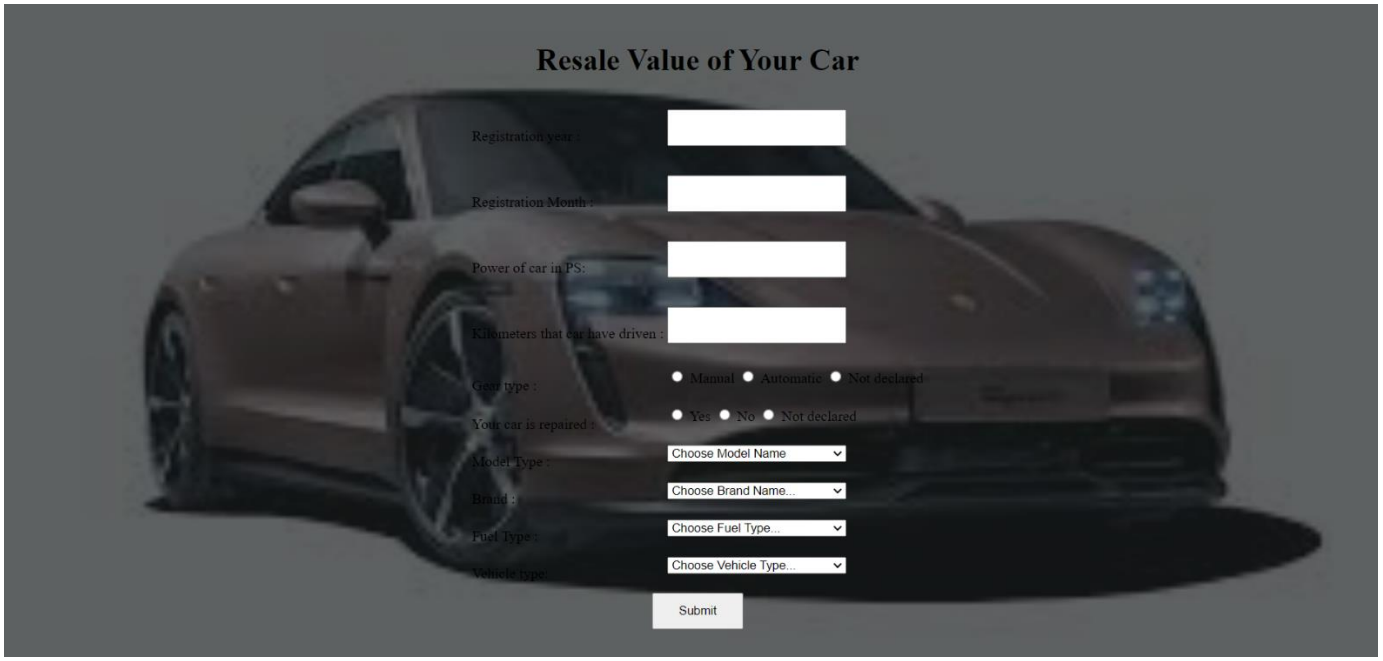
```
  for i in labels:
    mapper[i] = LabelEncoder()
    mapper[i].classes = np.load('../Result/'+str('classes'+i+'.npy'), allow_pickle=True)
    transform = mapper[i].fit_transform(new_df[i])
    new_df.loc[:,i+'_labels'] = pd.Series(transform, index=new_df.index)

  labeled = new_df[['yearOfReg','powerPS','kilometer','monthOfRegistration'] + [x+'_labels' f
or x in labels]]

  X = labeled.values.tolist()
  print('\n\n', X)
  #predict = reg_model.predict(X)

  # NOTE: manually define and pass the array(s) of values to be scored in the next line
  payload_scoring = {"input_data": [{"fields": [['yearOfReg', 'powerPS', 'kilometer', 'monthOf
Registration','gearbox_labels', 'notRepairedDamage_labels', 'model_labels','brand_labels', 'fue
lType_labels', 'vehicletype_labels']], "values": X}]}

  response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/7f67cbed-6222-413b-9901-
b2a72807ac82/predictions?version=2022-10-
30', json=payload_scoring, headers={'Authorization': 'Bearer ' + mltoken})
  predictions = response_scoring.json()
  print(response_scoring.json())
  predict = predictions['predictions'][0]['values'][0][0]
  print("Final prediction :",predict)

  return render_template('predict.html',predict=predict)

if __name__=='__main__':
  reg_model = load_model()#load the saved model
  app.run(host='localhost', debug=True, threaded=False)
```

8. TESTING:

Front Page of the Website



Entering the Details of the Car



Predicting the Value of the Car

9. RESULTS

9.1 PERFORMANCE METRIC

Here we will be evaluating the model built. We will be using the test set for evaluation. The test set is given to the model for prediction and prediction values are stored in another variable called y_pred. The r2 score of the model is calculated and its performance is estimated.

```
y_pred = regressor.predict(X_test)
print(r2_score(Y_test,y_pred))
```

Different regression models can be used to know the performance and choose whichever works better.

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES –

- Good at learning complex & non-linear relationships.
- Highly explainable & easy to interpret.
- Robust to others.
- No feature scaling is required.

DISADVANTAGES

- Consumes more time.
- Requires high computational power.

11. CONCLUSION

We started with understanding the use case of machine learning in the Automotive industry & how machine learning has transformed the driving experience. Moving on, we looked at how the various factors that the resale value of a car takes into consideration and performed a Explanatory Data Analysis (EDA). We believe that through the project of the Car Resale Value Prediction, it would be beneficial to many and help in the overall use and finding out the accurate pricing of the car that the customer is planning to buy to be to be found correctly.

12. FUTURE WORK

Only after getting proper feedback from the customers will we be able to make a proper development and update in the project, but we do have a few ideas like –

- Making the app area-friendly
- Help in making the customer meet the previous owner more efficiently
- Help customer select the type of car they want
- Help customer choose the brand they want

These are some of the few ideas we have in mind and will also put some new ideas after customer feedback.

13. APPENDIX

HTML FILES:

Car.html

```
<html>
<head>
<title>Car resale value </title>
<link rel="stylesheet" href="../StyleSheets/css/style.css">
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
</head>
<body>
<section class="header">

<div class="text-box">
<h1>Car resale value Predictor</h1>
<p>Predict the car resale value based on the arguments from the user</p>
<a href="./predict_page" class="visit-btn ">Check price</a>
</div>
</section>
</body>
</html>
```

Value.html

```
<html lang="en" dir="ltr">
<head>
```

```
<link rel="stylesheet" href="../StyleSheets/css/value.css">
<title>Car resale value</title>
</head>
<body>
<section class="form">
<form action="http://localhost:5000/predict" method="GET">
<table border="0" align="center">
<tbody>
<h1>Resale Value of Your Car</h1>
<tr>
<td><label for="year" padding:10px>Registration year : </label></td>
<td><input id="year" maxlength="50" name="regyear" type="text" />       <br><br>
</td>
</tr>
<tr>
<td><label for="month">Registration Month : </label></td>
<td><input id="month" maxlength="50" name="regmonth" type="text" /><br>
<br>
</td></tr>
<tr>
<td><label for="power">Power of car in PS: </label></td>
<td><input id="power" maxlength="50" name="powerps" type="text" />
<br><br></td></tr><tr>
<td><label for="kilometer">Kilometers that car have driven : </label></td>
<td><input id="kilometer" maxlength="50" name="kms" type="text" />
<br>
<br>
</td>
</tr>
<tr>
```

```
<td><label for="geartype">Gear type : </label></td>
<td><input type="radio" name="geartype" value="manual"/> Manual
<input type="radio" name="geartype" value="automatic"/> Automatic
<input type="radio" name="geartype" value="not-declared"/> Not declared
<br>
<br>
</td>
</tr>
<tr>
<td><label for="damage">Your car is repaired : </label></td>
<td><input type="radio" name="damage" value="yes"/> Yes
<input type="radio" name="damage" value="no"/> No
<input type="radio" name="damage" value="not-declared"/> Not declared
<br><br></td></tr><tr><td><label for="model">Model Type : </label></td><td>
<select name="model" id="model">
<option value="" disabled selected hidden>Choose Model Name</option>
<option value="golf">Golf </option>
<option value="grand">Grand </option>
<option value="fabia">Fabia </option>
<option value="3er">3er </option>
<option value="2_reihe">2 Reihe </option>
<option value="andere">Andere </option>
<option value="c_max">C Max </option>
<option value="3_reihe">3 Reihe </option>
<option value="passat">Passat </option>
<option value="navara">Navara </option>
<option value="ka">Ka </option>
<option value="polo">Polo </option>
<option value="twingo">Twingo </option>
<option value="a_klasse">A klasse </option>
```

```
<option value="scirocco">Scirocco </option>
<option value="5er">5er </option>
<option value="meriva">Meriva </option>
<option value="arosa">Arosa </option>
<option value="c4">C4 </option>
<option value="civic">Civic </option>
<option value="transporter">Transporter </option>
<option value="punto">Punto </option>
<option value="e_klasse">E Klasse </option>
<option value="clio">Clio </option>
</select> <br><br>
</td></tr>
<tr>
<td><label for="brand">Brand :</label></td>
<td>
<select name="brand" id="brand">
<option value="" disabled selected hidden>Choose Brand Name...</option>
<option value="volkswagen">Volkswagen </option>
<option value="audi">Audi </option>
<option value="jeep">Jeep </option>
<option value="skoda">Skoda </option>
<option value="bmw">Bmw </option>
<option value="peugeot">Peugeot </option>
<option value="ford">Ford </option>
<option value="mazda">Mazda </option>
<option value="nissan">Nissan </option>
<option value="renault">Renault </option>
<option value="mercedes_benz">Mercedes Benz </option>
<option value="opel">Opel </option>
<option value="seat">Seat </option>
```

```
<option value="citroen">Citroen </option>
<option value="honda">Honda </option>
</select>
<br><br></td></tr>
<tr>
<td><label for="fuelType">Fuel Type :</label></td>
<td>
<select name="fuelType" id="brand">
<option value="" disabled selected hidden>Choose Fuel Type...</option>
<option value="petrol"> Petrol </option>
<option value="diesel"> Diesel </option>
<option value="not-declared"> Not Declared </option>
<option value="lpg">LPG </option>
<option value="cng">CNG </option>
<option value="hybrid">Hybrid </option>
<option value="others">Others </option>
<option value="electric">Electric </option>
</select>
<br>
<br>
</td>
</tr>
<tr>
<td><label for="vehicletype">Vehicle type:</label></td>
<td>
<select name="vehicletype" id="vehicle" >
<option value="" disabled selected hidden>Choose Vehicle Type...</option>
<option value="coupe">Coupe </option>
<option value="suv">SUV </option>
<option value="kleinwagen">Kleinwagen </option>
```

```
<option value="limousine">Limousine </option>
<option value="cabrio">Cabrio </option>
<option value="bus">Bus </option>
<option value="kombi">Kombi </option>
<option value="andere">Andere </option>
<option value="volkswagen">Volkswagen </option>
</select>
<br>
<br>
</td>
</tr>
</tbody>
</table>
<input name="Submit" type="Submit" value="Submit" id="button"/>
</form>
</section>
</body
</html>
```

Predict.html

```
<html>
<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">
<link rel="stylesheet" href="../StyleSheets/css/predict.css">
<title>Car Resale Predicted Value</title>
</head>
<body>
<section class="header">
```

```
<nav>
<a href="/"><img src="../StyleSheets/Images/car1.jpg" width="100" height="100"></a>
</nav>
<div class="text-box">
<h1>The Predicted Car Resale Value is </h1>
<h1>{{predict}}</h1>
</div>
</section>
</body>
</html>
```

CSS FILES:

Style.css

```
*{
  margin: 0;
  padding: 0;
}

.header{
  min-height: 100vh;
  width: 100%;
  background-image: linear-gradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car1.jpg);
  background-position: center;
  background-size: cover;
  position: relative;
}
nav{
  display:flex;
```

```
  padding: 2% 6%;
  justify-content: space-between;
  align-items: center;
}
.nav-links{
  flex: 1;
  text-align: right;
}
.nav-links ul li{
  list-style: none;
  display: inline-block;
  padding: 8px 12px;
  position: relative;
}
.nav-links ul li a{
  color:white;
  text-decoration: none;
  font-size: 13px;
}
.text-box{
  text-align: center;
  position: relative;
  color:  #FFE4C4;
  top:50%;
}
.text-box h1{
  margin-top: 50px;
  font-size: 55px;
}
.text-box p{
```

```
  margin: 10px 0 40px;
  font-size: 15px;
}
.visit-btn{
  display: inline;
  border: 3px solid #fff;
  padding:10px 14px;
  font-size: 15px;
  background: transparent;
  color: white;
  text-decoration:none;
}
```

Value.css

```
.header{
  width: 100%;
  text-align: center;
  //padding-top: 20px;
  font-size:20px;
  font-family: "Lucida Console";
  background-color:#43FFB6;
  border:0%;
  top:0px;
  bottom:0px;
  right:0px;
  left:0px;
  overflow-y:auto;
}

body{
```

```
        margin: 0;
}
.form{
background-image: linear-gradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car4.png);
 background-position: center;
  background-size: cover;
  position: relative;
}
.form{
text-align: center;
padding:20px;
text-top:10px;
display: flex;
flex-direction: column;
align-items: center;
}
.form{
font-size:22px;
}
textarea {
  width: 100%;
  height: 150px;
  padding: 12px 20px;
  box-sizing: border-box;
  border: 2px solid #ccc;
  border-radius: 4px;
  background-color: #f8f8f8;
  resize: none;
}
input[type=text] {
```

```
  transition: width 0.4s ease-in-out;
}

input[type=text] {
  width: 70%;
  height: 10%;
  padding: 10px 10px;
  margin: 5px 0;
}
#model{
 width: 70%;
}
#brand{
width:70%;
}
#vehicle{
width:70%;
}
*{
color:black;
}

#button{
  padding: 10px 10px;
  margin: 0;

  text-align:center;
  width:100px;
}
```

Predict.css

```
.header{
  min-height: 100vh;
  width: 100%;
  background-image: linear-gradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car6.jpg);
  background-position: center;
  background-size: cover;
  position: relative;
}

.text-box{
  text-align: center;
  position: relative;
  color:  #FFE4C4;
  top:50%;
}
.text-box h1{
  margin-top: 50px;
  font-size: 55px;
}

.text-box p{
  margin: 10px 0 40px;
  font-size: 15px;
}

body{
        margin: 0;
}
```

```
nav{
  display:flex;
  padding: 2% 6%;
  justify-content: space-between;
  align-items: center;
}
```