A Project Report

on

CAR RESALE VALUE PREDICTION

TEAM ID: PNT2022TMID16122

Submitted by,

TEAM LEADER,

ABINESH R (927619BIT4001)

TEAM MEMBERS,

KALAIVAANI K S (927619BIT4043)

LAKSHANA S (927619BIT4051)

MRUDHHULA V S (927619BIT4058)

TABLE OF CONTENTS

CHAPTER - 1
INTRODUCTION

1.1 PROJECT OVERVIEW

Predicting the resale value of a car is not a simple task. It is trite knowledge that the value of used cars depends on a number of factors. The most important ones are usually the age of the car, its make (and model), the origin of the car (the original country of the manufacturer), its mileage (the number of kilometres it has run) and its horsepower. Due to rising fuel prices, fuel economy is also of prime importance. Unfortunately, in practice, most people do not know exactly how much fuel their car consumes for each km driven. Other factors such as the type of fuel it uses, the interior style, the braking system, acceleration, the volume of its cylinders (measured in cc), safety index, its size, number of doors, paint color, weight of the car, consumer reviews, prestigious awards won by the car manufacturer, its physical state, whether it is a sports car, whether it has cruise control, whether it is automatic or manual transmission, whether it belonged to an individual or a company and other options such as air conditioner, sound system, power steering, cosmic wheels, GPS navigator all may influence the price as well. Some special factors which buyers attach importance in Mauritius is the local of previous owners, whether the car had been involved in serious accidents and whether it is a lady-driven car. The look and feel of the car certainly contribute a lot to the price. As we can see, the price depends on a large number of factors.

1.2 PURPOSE

Car resale value prediction helps the user to predict the resale value of the car depending upon various features like kilo-metres 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. The most essential elements for forecast are brand and model, period use of vehicle, mileage of vehicle, gear type and fuel type utilised in the vehicle just as fuel utilisation per mile profoundly influences cost of a vehicle because of continuous changes in the cost of a fuel. In view of the differing highlights and factors, and furthermore with the assistance of master information the vehicle value forecast has been done precisely. This can enable the customers to make decisions. As a seller, he/she wants to get the maximum price but the aim is the opposite for the buyer or the car dealer. So, to become aware of such things should be given importance. Need to calculate resale value of the car with the help of the correct valuation tool to know the market price or what could be the market price of the vehicle. To negotiate with the dealer or seller with due diligence and end up in a profitable deal. Estimating the best price for the car. Getting insight into industry rates and trends. Safeguarding against underhanded practices. Confidence for negotiations.

CHAPTER - 2
LITERATURE SURVEY

2.1 EXISTING PROBLEM

The prices of new cars in the industry are 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

2.2.1 TITLE- USED CAR PRICE PREDICTION
AUTHOR- Praful Rane1, Deep Pandya2, Dhawal Kotak3
DESCRIPTION

The price of a new car 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 prices of new cars and the financial incapability of the customers to buy them, Used Car sales are on a global increase In fact, seller also has no idea about the car's existing value or the price he should be selling the car at. To overcome this problem, we have developed a model which will be highly effective. Regression Algorithms are used because they provide us with continuous value as an output and not a categorized value. Because of which it will be possible to predict the actual price a car rather than the price range of a car. User Interface has also been developed which acquires input from any user and displays the Price of a car according to user's inputs.

2.2.2 TITLE- Vehicle Price Prediction using SVM Techniques
AUTHOR- S.E. Viswapriya, Darbuka Sai Sandeep Sharma, Gandavarapu Sathya Kiran
DESCRIPTION

The prediction of price for a vehicle has been more popular in research area, and it needs predominant effort and information about the experts of this particular field. The number of different attributes is measured and also it has been considerable to predict the result in more reliable and accurate. To find the price of used vehicles a well-defined model has been developed with the help of three machine learning techniques such as Artificial Neural Network, Support Vector Machine and Random Forest. These techniques were used not on the individual items but for the whole group of data items. This data group has been taken from some web portal and that same has been used for the prediction. The data must be collected using web scraper that was written in PHP programming language. Distinct machine learning

algorithms of varying performances had been compared to get the best result of the given data set. The final prediction model was integrated into Java application.

2.2.3 TITLE- Used Cars Price Prediction and Valuation using Data Mining Techniques
AUTHOR- Abdulla AlShared
DESCRIPTION

Due to the unprecedented number of cars being purchased and sold, used car price prediction is a topic of high interest. Because of the affordability of used cars in developing countries, people tend more purchase used cars. A primary objective of this project is to estimate used car prices by using attributes that are highly correlated with a label (Price). To accomplish this, data mining technology has been employed. Null, redundant, and missing values were removed from the dataset during pre-processing. In this supervised learning study, three regressors (Random Forest Regressor, Linear Regression, and Bagging Regressor) have been trained, tested, and compared against a benchmark dataset. Among all the experiments, the Random Forest Regressor had the highest score at 95%, followed by 0.025 MSE, 0.0008 MAE, and 0.0378 RMSE respectively. In addition to Random Forest Regression, Bagging Regression performed well with an 88% score, followed by Linear Regression having an 85% mark. A train-test split of 80/20 with 40 random states was used in all experiments. The researchers of this project anticipate that in the near future, the most sophisticated algorithm is used for making predictions, and then the model will be integrated into a mobile app or web page for the general public to use.

2.2.4 TITLE- Prediction of Resale Value of the Car Using Linear Regression Algorithm
AUTHOR- Kiran S
DESCRIPTION

A correlation with each attribute to that of target attribute is found and linear regression curve with the target attribute is drawn. As a final step the total error and accuracy is measured. The demand for used cars has increased significantly in the past decade and it is prognosticated that with Covid-19 outbreak this requirement will augment considerably. Hence to enhance the reliability, with the expansion of the used car market, a model that can forecast the current market price of a used automobile on the basis of a variety of criteria. This analysis can be used to study the trends in the industry, offer better insight into the market, and aid the community in its smooth workflow. The aim of this research paper is to predict the car price as per the data set (previous consumer data like engine capacity, distance travelled, year of manufacture, etc.). The result of these algorithms will be analysed and based on the efficiency and accuracy of these algorithms, the best one of them can be used for the said purpose.

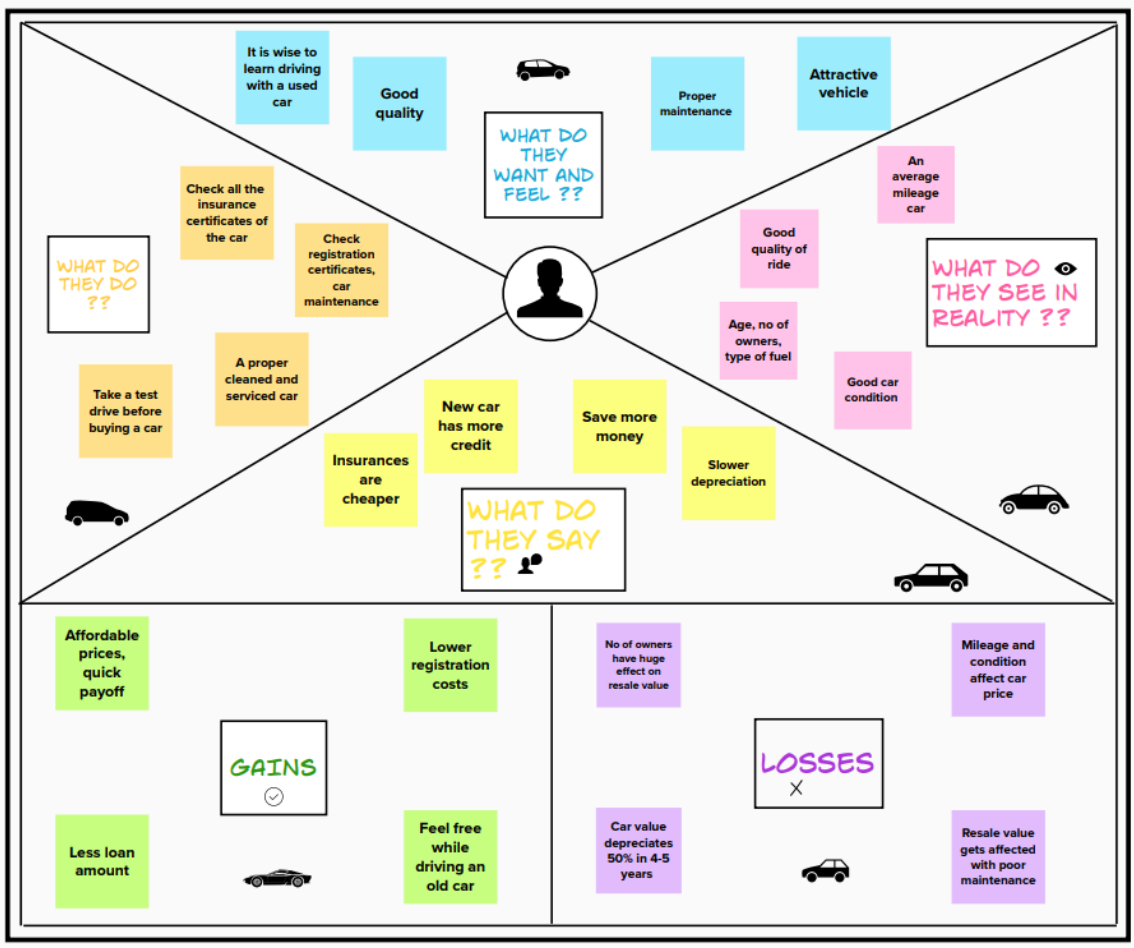
2.3 PROBLEM STATEMENT DEFINITION

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. Therefore, we need an intelligent solution to predict the accurate resale value of the car and present it to the users in a web application.

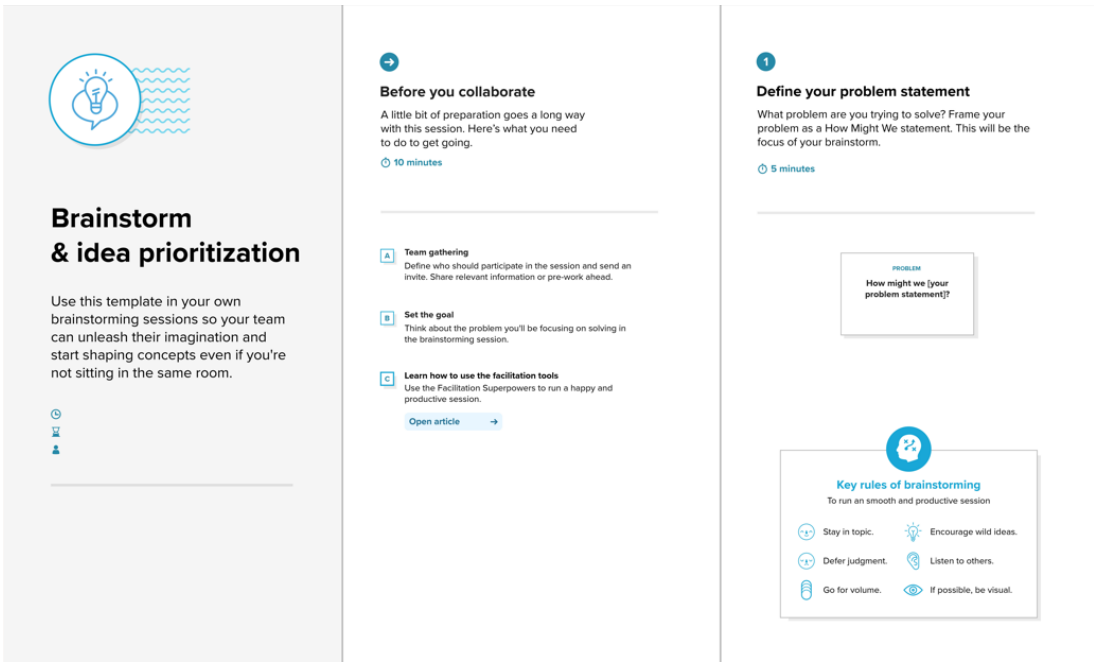
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Owner of the car	Sell the car	Don't know its resale value	Of wear and tear over the years (damage due to daily use)	Worried
PS-2	Owner of the car (high priced /luxury car)	Find resale value of the car	In a dilemma of finding correct resale value	Of high price and high maintenance cost	Confused
PS-3	Owner of the car	Sell the car	In urge of selling the car in a satisfactory price	Urgent financial requirements	Frustrated

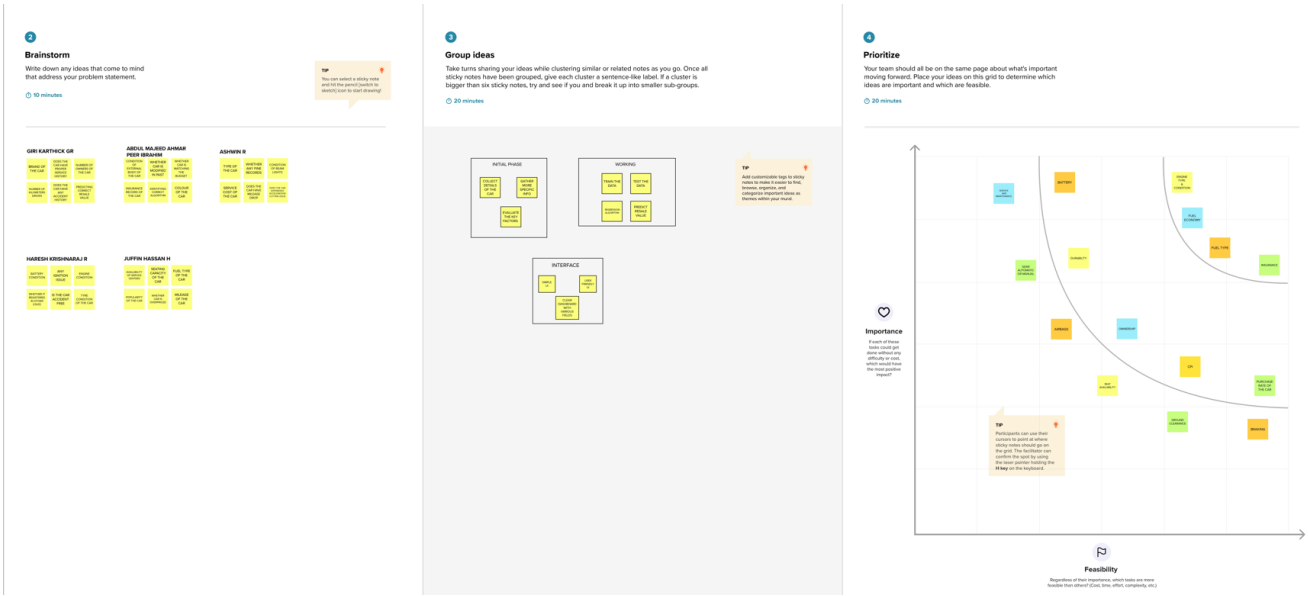
CHAPTER - 3
IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING





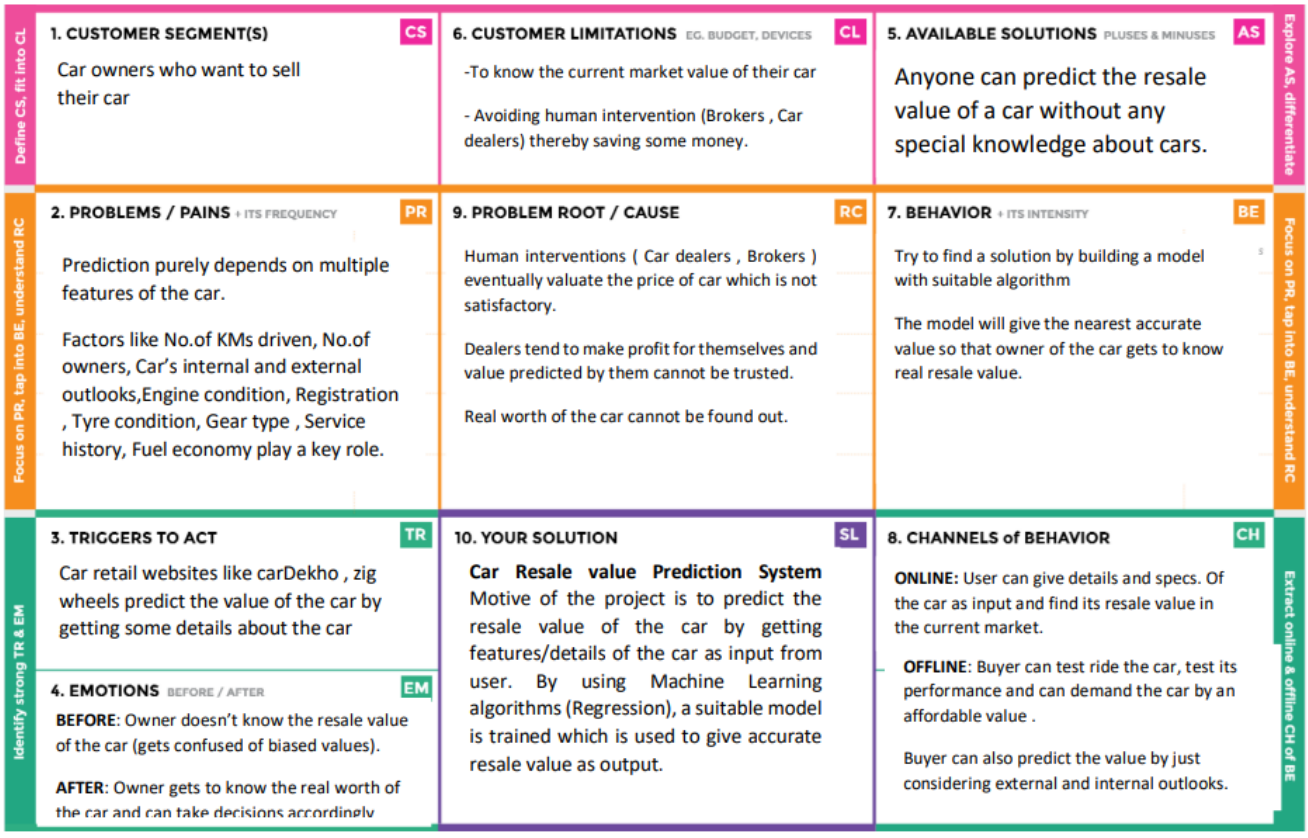
3.3 PROPOSED SOLUTION

S. No.	Parameter	Description
1	Problem Statement (Problem to be solved)	To develop a web application to predict the resale value of a car based on its features
2	Idea / Solution description	To train the system with the dataset and train them using a regression model
3	Novelty / Uniqueness	Predict more accurate value based on the input given by the user.
4	Social Impact / Customer Satisfaction	Customer can get a clear recommendation of the resale value of the car and based on the evaluation; customer can take decisions on selling the car.
5	Business Model (Revenue Model)	Free and can be accessed by anyone. Prediction purely depends on features of the car.
6	Scalability of the Solution	More the specific details, more the accuracy. An accurate resale value is predicted by the trained model.

3.4 PROBLEM SOLUTION FIT

The sales of second-hand imported cars and used cars is increasing nowadays. Predicting the price of used cars is an important and interesting problem. Predicting the resale value of a car is not an easy task. It is trite knowledge that the value of used cars depends on a number of factors. The value of a car drops right from the moment it is bought and the depreciation continues with each passing year. In fact, in the first year itself, the value of a car decreases by 20 percent of its initial value. The make and model of a car, total kilometres driven, overall condition of the car and various other factors further affect the car's resale value. So, it is necessary to build a model and design an application or website to estimate the price of used cars. The model should take car related parameters and output a selling price of the car. The selling price of a used car depends on certain features as mentioned below:

- Fuel Type
- Manufacturing Year
- Miles Driven
- Number of Historical Owners
- Maintenance Record, etc.,



CHAPTER – 4
REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

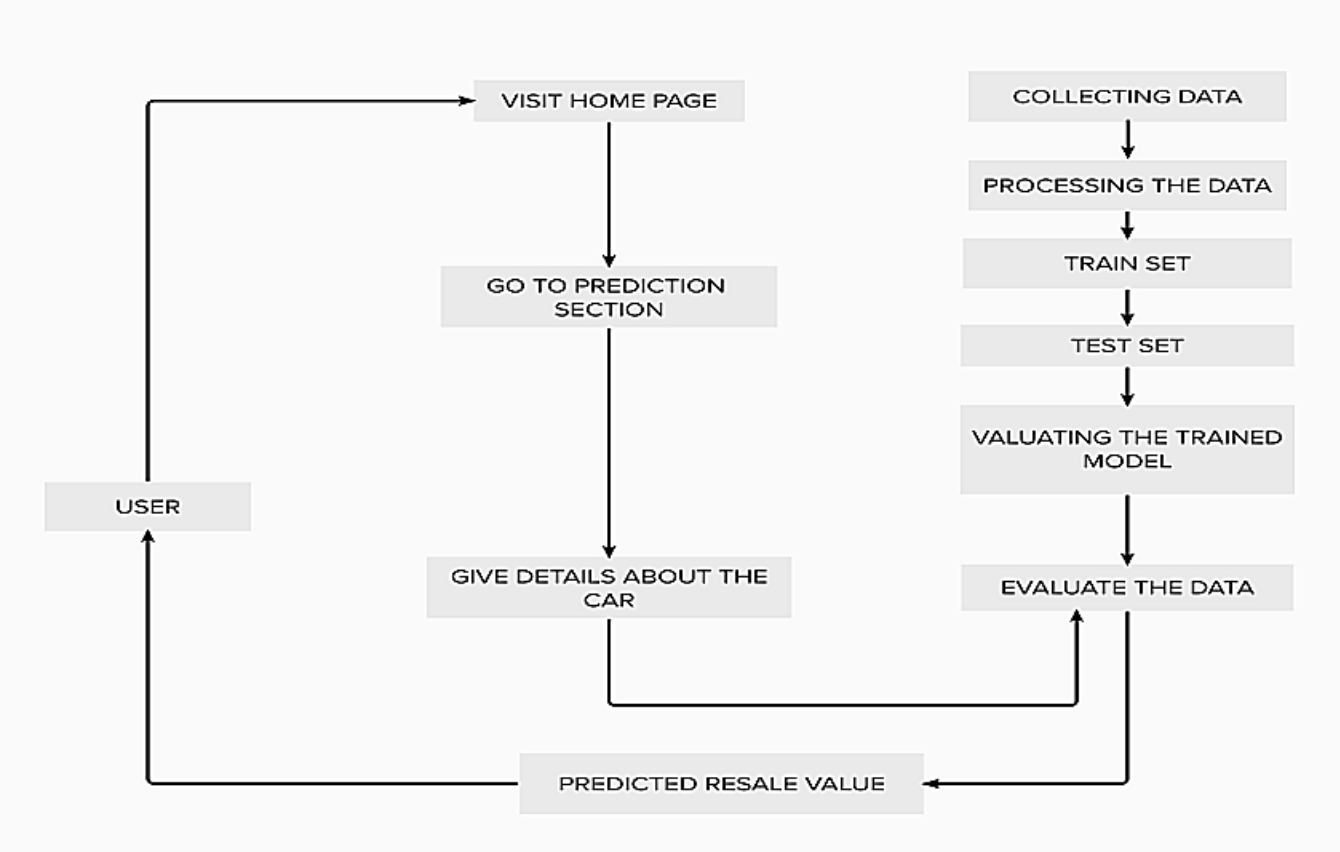
FR No	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Opens Website	Server Opens the site
FR-2	User Enters the Values	User enters Car details
FR-3	Validation	User Values Checks and validation
FR-4	Prediction	Car Price Prediction

4.2 NON-FUNCTIONAL REQUIREMENTS

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	A Simple, User-friendly user interface which can give solution to the user. (i.e resale value
NFR-3	Reliability	Reliable and consistent prediction can be done.
NFR-4	Performance	Performance does not depend on amount of end users. It is scalable, consistent with no delay.
NFR-5	Availability	Uninterrupted services must be available other than any server issues.
NFR-6	Scalability	It can handle any amount of data irrespective of number of users. It can perform computations in a fast & cost-effective manner.

CHAPTER - 5
PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

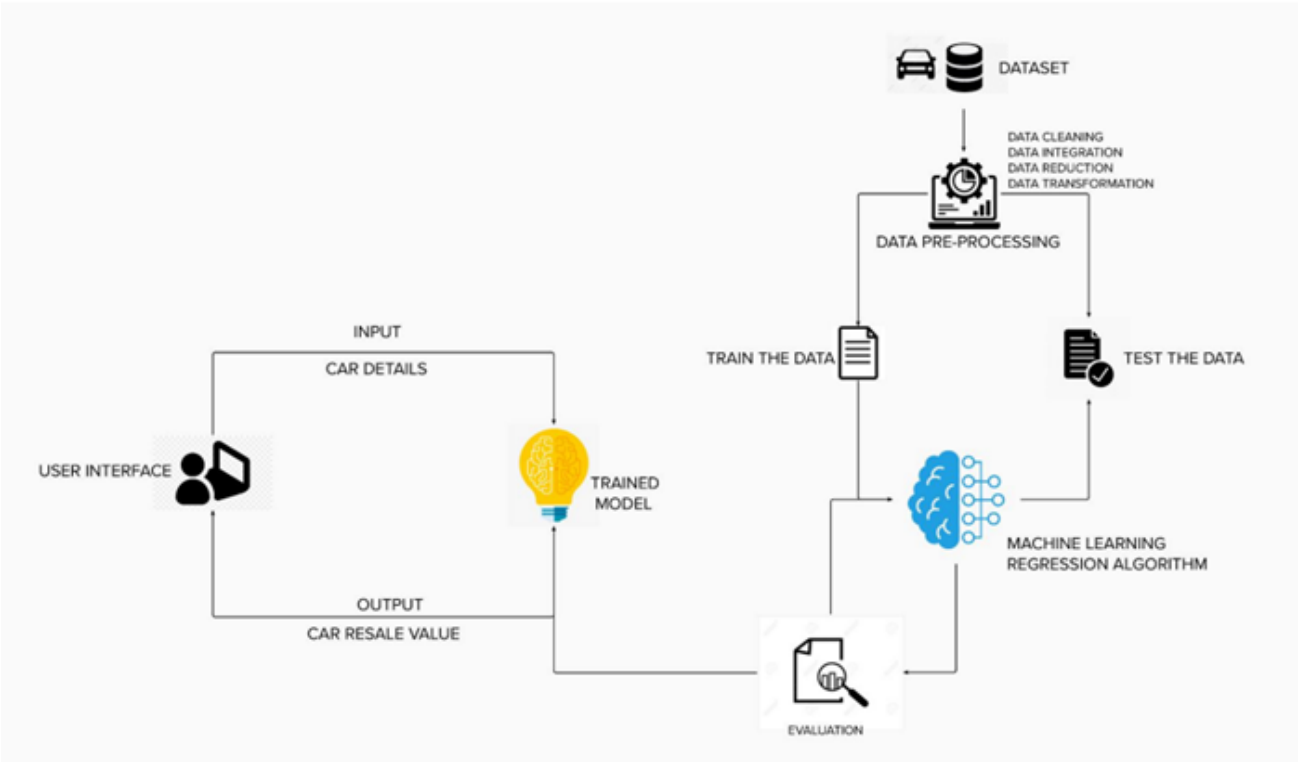


5.2 SOLUTION & TECHNICAL ARCHITECTURE

Solution Architecture is an architectural description of a specific solution. It is the practice of designing, describing, and managing solution engineering to match its specific business problems. It comprises subprocesses that draw guidance from various enterprise architecture viewpoints. Solution Architecture is the initial step taken when an organization aims to create a set of enterprise solutions, applications and processes that integrate with each other in order to address specific needs and requirements and that often lead to software architecture and technical architecture work.

Technical architecture—which is also often referred to as application architecture, IT architecture, business architecture, etc.—refers to creating a structured software solution that will meet the business needs and expectations while providing a strong technical plan for the growth of the software application through its lifetime. IT architecture is equally important to the business team and the information technology team. Technical architecture includes the major components of the system, their relationships, and the contracts that define the interactions between the components. The goal of technical architects is to achieve all the business needs with an application that is optimized for both performance and security. The technical architecture of the proposed project is as follows:

S. No	Component	Technology
1	User Interface	HTML, CSS, JavaScript
2	Application Logic-1	Python - Flask
3	Application Logic-2	IBM Watson STT service
4	Machine learning	Regression, Random Forest



5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (user)	Visit the website	Usn-1	As a user, i can visit the home page of the website and get to know about the website.	I can access the website by correct URL.	High	Sprint-1
	Enter the condition	USN-2	User can enter the car conditions.	User can able to enter the details	High	Sprint-1
	Predict the value of my old car	USN-3	As a user, I can predict the resale Value of the car by giving its details/ Features.	I can enter car details	High	Sprint-2
	Look for a price of a used car	USN-4	As a user and also a buyer or Seeker, I can also get to know the Price of the used car.	I can choose most likely car.	Medium	Sprint-3
Administrator	Application	USN-5	As an administrator I can upgrade / update /modify the application with additional features.		Medium	Sprint-4

CHAPTER - 6
PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint planning is an event in scrum that kicks off the sprint. The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved. Sprint planning is done in collaboration with the whole scrum team. In scrum, the sprint is a set period of time where all the work is done. However, before you can leap into action you have to set up the sprint. You need to decide on how long the time box is going to be, the sprint goal, and where you're going to start. The sprint planning session kicks off the sprint by setting the agenda and focus. If done correctly, it also creates an environment where the team is motivated, challenged, and can be successful. Bad sprint plans can derail the team by setting unrealistic expectations. The following is the sprint planning and estimation for the project

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Build flask web page	USN-1	Create flask web page	10	High	ABINESH R MURUDHULA V S
Sprint-1	Integrate with model	USN-2	Get inputs and integrate with model and check the outputs	10	High	LAKSHANA S KALAIVAANI K S
Sprint-2	Building the Model	USN-3	Building a python code for predicting price	5	Medium	ABINESH R MURUDHULA V S
Sprint-2	Deploy with Data set	USN-4	Predict the data using the model	10	Medium	LAKSHANA S KALAIVAANI K S
Sprint-3	Build Model in Cloud	USN-5	Integrate Web flask and Data models and check the integrity's	20	High	ABINESH R MURUDHULA V S LAKSHANA S KALAIVAANI K S
Sprint-4	Display the predict value	USN-6	Finally, Integrate Flask with Scoring End Point	20	High	LAKSHANA S KALAIVAANI K S

6.2 SPRINT DELIVERY SCHEDULE

Since sprints take place over a fixed period of time, it's critical to avoid wasting time during planning and development. And this is precisely where sprint scheduling enters the equation.In case you're unfamiliar, a sprint schedule is a document that outlines sprint planning from end to end. It's one of the first steps in the agile sprint planning process—and something that requires adequate research, planning, and communication. Sprint Delivery Schedule for the proposed solution

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	25 Oct 2022
Sprint-2	15	6 Days	31 Oct 2022	05 Nov 2022	15	03 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	07 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	16 Nov 2022

CHAPTER - 7
CODING & SOLUTIONING

7.1 FEATURE 1

Developing the model and building the application using Random Forest Regression With greater accuracy the model building will be possible with building the model with pickle setup which is used with label encoder to get the desired integer labels for the string inputs. It's very much effective method will get better accuracy the linear regression methods.

```
import pandas as pd
import numpy as np
import matplotlib as plt
from sklearn.preprocessing import LabelEncoder
import pickle
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3
import io
def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='B__tpAmla-ENJ92sDpmg_feytHFBlctw-IrdKxSRqx2w',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'carresalevalueprediction-donotdelete-pr-u86776xsgyuhtj'
object_key = 'autos.csv'

body = cos_client.get_object(Bucket=bucket,Key=object_key)
df = pd.read_csv((io.BytesIO(body['Body'].read())) , header=0 , sep=',' ,encoding='Latin1',low_memory=False)
df.head()

print(df.seller.value_counts())
df[df.seller !='gewerblich']
df=df.drop('seller',axis=1)

print(df.offerType.value_counts())
df[df.offerType !='Gesuch']
df=df.drop('offerType',axis=1)

print(df.shape)
df=df[(df.powerPS>50) & (df.powerPS<900)]
print(df.shape)
df=df[(df.yearOfRegistration>=1950)&(df.yearOfRegistration<2022)]
print(df.shape)
```

```
df.drop(['name','abtest','dateCrawled','nrOfPictures','lastSeen','postalCode','dateCreated'],
axis='columns',inplace=True)

new_df=df.copy()
new_df=new_df.drop_duplicates(['price','vehicleType','yearOfRegistration','gearbox','powerP
S','model','kilometer','monthOfRegistration','fuelType','notRepairedDamage'])

new_df.gearbox.replace(('manuell','automatik'),('manual','automatic'),inplace=True)
new_df.fuelType.replace(('benzin','andere','elektro'),('petrol','others','electric'),inplace=True)
new_df.vehicleType.replace(('kleinwagen','cabrio','kombi','andere'),('samll
car','convertible','combination','others'),inplace=True)
new_df.notRepairedDamage.replace(('ja','nein'),('Yes','No'),inplace=True)

new_df=new_df[(new_df.price>=100)&(new_df.price<=150000)]

new_df['notRepairedDamage'].fillna(value='not-declared',inplace=True)
new_df['fuelType'].fillna(value='not-declared',inplace=True)
new_df['gearbox'].fillna(value='not-declared',inplace=True)
new_df['vehicleType'].fillna(value='not-declared',inplace=True)
new_df['model'].fillna(value='not-declared',inplace=True)

from ibm_watson_machine_learning import APIClient
wml_credentials={
   "url":"https://us-south.ml.cloud.ibm.com",
   "apikey":"MGU1iT6RDkhiyFrQhD8KbdYD1kWSOWNmSZCUhCB_IGDg"
}
client =APIClient(wml_credentials)

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

space_uid=guide_from_space_name(client,'CarPrice')
print("Space UID"+ space_uid)

client.set.default_space(space_uid)

client.software_specifications.list()

software_spec_uid  = client.software_specifications.get_uid_by_name("runtime-22.1-py3.9")
software_spec_uid

labels=['gearbox','notRepairedDamage','model','brand','fuelType','vehicleType']

mapper={}
for i in labels:
   mapper[i]=LabelEncoder()
   mapper[i].fit(new_df[i])
```

```
    tr=mapper[i].transform(new_df[i])
    np.save(str('classes'+i+'.npy'),mapper[i].classes_)
    print(i,":",mapper[i])
    new_df.loc[:, i+ '_labels']=pd.Series(tr,index=new_df.index)

labeled = new_df[['price','yearOfRegistration','powerPS','kilometer','monthOfRegistration']+[x+"_labels" for x in labels]]
print(labeled.columns)

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)

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'))

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

model_details = client.repository.store_model(model=regressor,meta_props={
    client.repository.ModelMetaNames.NAME: "resale_model",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_spec_uid,
    client.repository.ModelMetaNames.TYPE: "scikit-learn_1.0"
})
model_id = client.repository.get_model_id(model_details)

model_id

X_train[0]

regressor.predict([[2022.0, 179.0, '1500', 12.0, 0, 0, 30, 1, 1, 4]])
```

Here we have implemented the pandas and numpy packages to read the input files as well to get the load and dump files in the pickle package. Here we have built the model with IBM Watson because we can’t use always the application in offline we need a online source that every user can use the prediction methods. Where IBM cloud will deploy the application and we can get better user results of price of used car.

7.2 FEATURE 2

Always the user can't come to the Coding part and comes to get the results because the user don't know about the coding applications so the User builds need interactive website which user can easily catch the results by just entering the car functionalities and just the details of car.

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

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "MGU1iT6RDkhiyFrQhD8KbdYD1kWSOWNmSZCUhCB_IGDg"
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(__name__,template_folder='templates/')
@app.route('/')
def index():
    return render_template('index.html')

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

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

@app.route('/y_predict',methods=['GET','POST'])
def y_predict():
    regyear = int(request.form['regyear'])
    powerps = float(request.form['powerps'])
    kms = float(request.form['kms'])
    regmonth = int(request.form.get('regmonth'))
    gearbox = request.form['gearbox']
    damage = request.form['dam']
    model  = request.form.get('model_type')
    brand = request.form.get('brand')
    fuelType = request.form.get('fuel')
    vehicletype= request.form.get('vehicletype')
    new_row =
{'yearOfRegistration':regyear,'powerPS':powerps,'kilometer':kms,'monthOfRegistration':regmonth,'gearbox':gearb
ox,'notRepairedDamage':damage,'model':model,'brand':brand,'fuelType':fuelType,'vehicleType':vehicletype}

    print(new_row)
    new_df =
pd.DataFrame(columns=['vehicleType','yearOfRegistration','gearbox','powerPS','model','kilometer','monthOfRegis
tration','fuelType','brand','notRepairedDamage'])
```

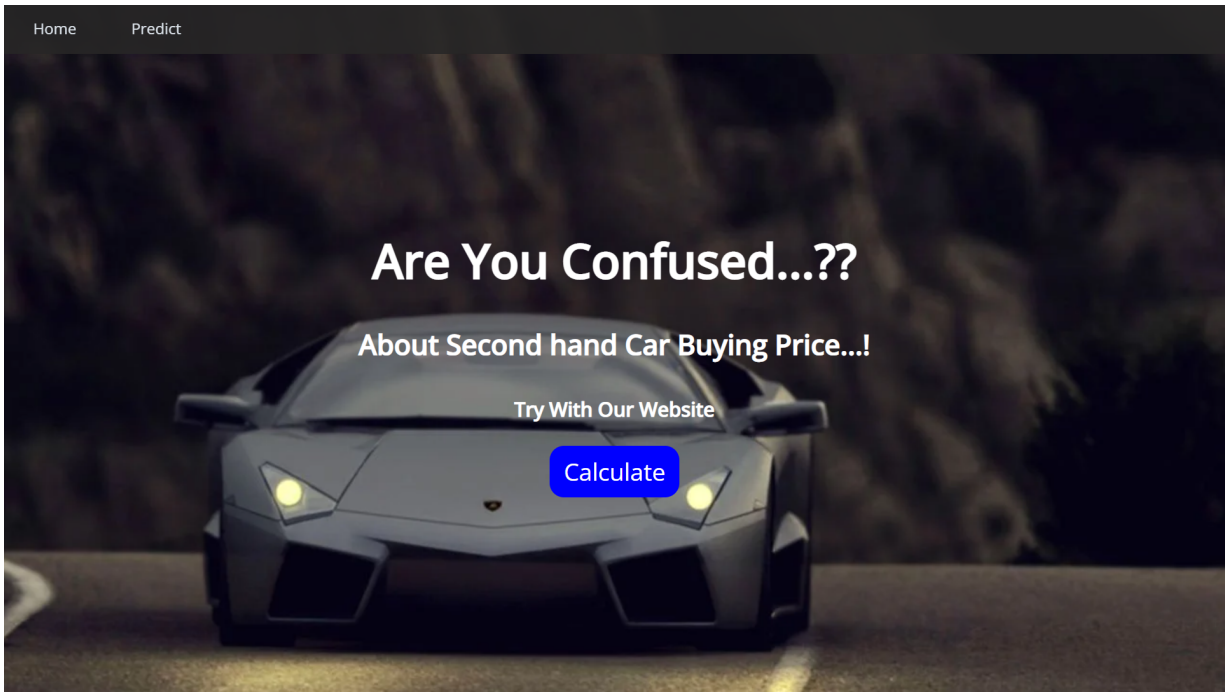
```
    new_df = new_df.append(new_row,ignore_index=True)
    labels = ['gearbox','notRepairedDamage','model','brand','fuelType','vehicleType']
    mapper = {}
    for i in labels:
        mapper[i] = LabelEncoder()
        mapper[i].classes_ = np.load(str('classes'+i+'.npy'),allow_pickle=True)
        tr = mapper[i].fit_transform(new_df[i])
        new_df.loc[:,i+'_Labels'] = pd.Series(tr,index=new_df.index)
    labeled = new_df[ ['yearOfRegistration','powerPS','kilometer','monthOfRegistration'] + [x+"_Labels" for x in
labels]]

    X = labeled.values
    print(X)

    payload_scoring = {"input_data": [{"field":
[['vehicleType','yearOfRegistration','gearbox','powerPS','model','kilometer','monthOfRegistration','fuelType','brand
','notRepairedDamage']], "values": X.tolist()}]}

    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/392de23a-3e4e-40ce-
85b0-b1accd4fddda/predictions?version=2022-11-17', json=payload_scoring,
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    predictions = response_scoring.json()
    print(predictions['predictions'][0]['values'][0][0])
    return render_template('resalepredict.html',ypred="{:.2f}".format(predictions['predictions'][0]['values'][0][0]))

if __name__ == '__main__':
    app.run(host='Localhost',debug=True,threaded=False)
```



Enter Your Car Details to Predict Price

Year Of Registration *	Month Of Registration *
Power of Car in PS *	Kilometers the Car has Driven *
Select Gear Box Type *	Your Car is damaged or repaired *
Model Type *	Brand of the Car *
Fuel type of the car *	Vehicle Type *

Predict The Price

THE RESALE VALUE PREDICTED IS

₹90492.40

CHAPTER - 8
TESTING

8.1 TEST CASES

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute
HOME_ PAGE_01	Functional	Home Page	User can See the Images and Hyper Links and visits the calculate Button	Internet Connectivity	1.Enter URL and click go 2.See the Images and Styles 3.Button to redirect page
PREDICTION_ PAGE_01	Functional	Predict	User can enter the values as their car condition	Car details	1.Enter URL Predict button and click go 2.User should enter the details 3. Verify UI elements: a. Text box b. Value scrolls c. Select options 3. User can click submit
RESULT PAGE	Functional	Result	User can able to see the car price		Only to see the Price

Test Data	Expected Result	Actual Result	Status	Commments	TC for Automation(Y/N)	Executed By
	Home page Should Open	Working as expected	Pass	Improving CSS	N	LAKSHANA
vehicleType','yearOfRegistration','gearbox','powerPS','model','kilometer','monthOfRegistration','fuelType','brand','notRepairedDamage'	1. User can able to see the input page. 2. User can enter the values. 3. Values able to enter	Working as expected	Pass	Nill	N	KALAIVAANI
	User can see the Car Resale Price	Working as expected	Pass	Nill	N	LAKSHANA

8.2 USER ACCEPTANCE TESTING

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	4	2	2	15
Duplicate	1	0	0	0	1
External	0	0	0	0	0
Fixed	5	4	1	1	11
Not Reproduced	0	0	0	0	0
Skipped	1	1	0	0	2
Won't Fix	0	0	0	0	0
Totals	14	9	3	3	29

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	5	0	0	5
Security	1	0	0	1
Final Report Output	6	0	0	6
Version Control	2	0	0	2

CHAPTER - 9
RESULTS

THE RESALE VALUE PREDICTED IS

₹90492.40

Fig : Project Output

9.1 PERFORMANCE METRICS

S. No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - , MSE - , RMSE - , R2 score - **Classification Model:** Confusion Matrix - , Accuray Score- & Classification Report -	{'mae': 1325.112086905962, 'mse': 9577053.62710202, 'rmse': 3094.6815065692977, 'rmsle': 8.03744027403009, 'r2': 0.8661221626879432, 'adj_r2_score': 0.8661152969113608}

2.	Tune the Model	Hyperparameter Tuning - Validation Method -	`y_pred = regressor.predict(X_test)`
			`print(r2_score(Y_test,y_pred))`
			`0.8310350387286918`

CHAPTER - 10
ADVANTAGES & DISADVANTAGES

ADVANTAGES

- **Used car sellers (dealers):** They are one of the biggest target groups that can be interested in results of this study. If used car sellers better understand what makes a car desirable, what the important features are for a used car, then they may consider this knowledge and offer a better service.

- **Online pricing services:** There are websites that offers an estimate value of a car. They may have a good prediction model. However, having a second model may help them to give a better prediction to their users. Therefore, the model developed in this study may help online web services that tells a used car's market value.

- **Individuals:** There are lots of individuals who are interested in the used car market at some points in their life because they wanted to sell their car or buy a used car. In this process, it's a big corner to pay too much or sell less then it's market value

- Able to give accurate and acceptable price for both buyer and seller.
- Have range of option on buying on budget.
- Helps in saving money than giving to brokerage.

DISADVANTAGES

- Poor checking and invalid information affect the value of prediction.
- Cars are limited usage vehicles some people only could afford this basis on knowledge-based purchasing.

CHAPTER - 11
CONCLUSION

In the given guided project, I understood the problem to classify if it is aregression or a classification kind of problem. I also came to know how to pre-process the data using different data pre-processing techniques. Not only this, I also grasp the knowledge about applying different algorithms according to the dataset. I also learn about the features of flask application

CHAPTER - 12
FUTURE SCOPE

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.

CHAPTER - 13
APPENDIX

13.1 SOURCE CODE

13.1.1 App.py

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

import requests

# NOTE: you must manually set API_KEY below using information retrieved from your IBM Cloud account.
API_KEY = "MGU1iT6RDkhiyFrQhD8KbdYD1kWSOWNmSZCUhCB_IGDg"
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(__name__,template_folder='templates/')
@app.route('/')
def index():
    return render_template('index.html')

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

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

@app.route('/y_predict',methods=['GET','POST'])
def y_predict():
    regyear = int(request.form['regyear'])
    powerps = float(request.form['powerps'])
    kms = float(request.form['kms'])
    regmonth = int(request.form.get('regmonth'))
    gearbox = request.form['gearbox']
    damage = request.form['dam']
    model  = request.form.get('model_type')
    brand = request.form.get('brand')
    fuelType = request.form.get('fuel')
```

```
    vehicletype= request.form.get('vehicletype')
    new_row =
{'yearOfRegistration':regyear,'powerPS':powerps,'kilometer':kms,'monthOfRegistration':regmonth,'gearbox':gearbo
x,'notRepairedDamage':damage,'model':model,'brand':brand,'fuelType':fuelType,'vehicleType':vehicletype}

    print(new_row)
    new_df =
pd.DataFrame(columns=['vehicleType','yearOfRegistration','gearbox','powerPS','model','kilometer','monthOfRegistr
ation','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(str('classes'+i+'.npy'),allow_pickle=True)
        tr = mapper[i].fit_transform(new_df[i])
        new_df.loc[:,i+'_Labels'] = pd.Series(tr,index=new_df.index)
    labeled = new_df[ ['yearOfRegistration','powerPS','kilometer','monthOfRegistration'] + [x+"_Labels" for x in
labels]]

    X = labeled.values
    print(X)

    payload_scoring = {"input_data": [{"field":
[['vehicleType','yearOfRegistration','gearbox','powerPS','model','kilometer','monthOfRegistration','fuelType','brand','
notRepairedDamage']], "values": X.tolist()}]}

    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/392de23a-3e4e-40ce-
85b0-b1accd4fddda/predictions?version=2022-11-17', json=payload_scoring,
    headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    predictions = response_scoring.json()
    print(predictions['predictions'][0]['values'][0][0])
    return render_template('resalepredict.html',ypred="{:.2f}".format(predictions['predictions'][0]['values'][0][0]))

if __name__ == '__main__':
    app.run(host='Localhost',debug=True,threaded=False)
```

index.html

```
<html>
<head>
  <title>CAR RESALE VALUE PREDICTION</title>
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link href="https://fonts.googleapis.com/css2?family=Open+Sans:wght@300;400;500&display=swap" rel="stylesheet">
  <style>
    body {
```

```
    font-family: 'Open Sans', sans-serif;
    margin: 0;
}

div.landing nav {
    background-color: rgb(39, 38, 38);
    width: 100%;
    position: fixed;
    opacity: 0.9;
    height: 6%;
    padding-bottom: 0.5em;
}

div.landing ul {
    padding: 0%;
    margin: 0%;
}

div.landing nav li {
    display: inline;
    padding: 0.2em;
    float: left;
}

div.landing nav li:hover {
    background-color: rgb(67, 66, 66);
    height: 100%;
}

div.landing nav a {
    text-decoration: none;
    display: inline-block;
    color: aliceblue;
    padding: 10px;
    width: 80px;
    text-align: center;

}

div.block1 {
    text-size-adjust: 100px;
    text-align: center;
    width: 100%;
    height: 100%;
    display: block;
    color: rgb(255, 255, 255);
    font-size: 25px;
    background-image: url(https://wallpaperaccess.com/full/1656595.jpg);
    background-position: center;
```

```
        }

        div.blockin {
            padding: 200px 150px;
        }

        button {
            color: blue;
        }

        .button {
            background-color: blue;
            border-radius: 15px;
            color: white;
            text-decoration: none;
            text-decoration: solid;
            padding: 10px 15px;
        }

        .button:focus,
        .button:hover {
            background-color: rgb(255, 0, 212);
            color: White;
        }
    </style>
</head>

<body>
    <div>
        <div class="landing" id="land">
            <nav>
                <ul>
                    <li>
                        <a href="index.html">Home</a>
                    </li>
                    <li>
                        <a href="resaleintro.html">Predict</a>
                    </li>
                </ul>
            </nav>
            <!--main header block-->
            <div class=block1 id="head">
                <div class="blockin">
                    <h1>Are You Confused...??</h1>
                    <h3>About Second hand Car Buying Price...!</h3>
                    <h5>Try With Our Website</h5>
                    <a href="resaleintro.html" class="button js-button" role="button">Calculate</a>
                </div>
            </div>
```

```
    </div>
  </div>
</body>
</html>
```

resaleintro.html

```
<!DOCTYPE html>
<html>

<head>
  <title> CAR RESALE VALUE PREDICTION</title>
  <style>
    @import url("https://fonts.googleapis.com/css?family=Open+Sans:400,700");

    :root {
      --white: #afafaf;
      --red: #e31b23;
      --bodyColor: #292a2b;
      --borderFormEls: hsl(0, 0%, 10%);
      --bgFormEls: hsl(0, 0%, 14%);
      --bgFormElsFocus: hsl(0, 7%, 20%);
    }

    * {
      padding: 0;
      margin: 0;
      box-sizing: border-box;
      outline: none;
    }

    a {
      color: inherit;
    }

    input,
    select,
    textarea,
    button {
      font-family: inherit;
      font-size: 100%;
    }

    button,
    label {
      cursor: pointer;
    }

    select {
```

```
    appearance: none;
}

/* Remove native arrow on IE */
select::-ms-expand {
    display: none;
}

select:-moz-focusring {
    color: transparent !important;
    text-shadow: 0 0 0 var(--white);
}

textarea {
    resize: none;
}

ul {
    list-style: none;
}

body {
    font: 18px/1.5 "Open Sans", sans-serif;
    background: var(--bodyColor);
    color: var(--white);
    margin: 1.5rem 0;
    text-align: center;

}

.container {

    max-width: 800px;
    margin: 0 auto;
    padding: 0 1.5rem;
}

.my-form h1 {
    margin-bottom: 1.5rem;
}

.my-form li,
.my-form .grid>*:not(:last-child) {
    margin-bottom: 1.5rem;
}

.my-form select,
.my-form input,
.my-form textarea,
```

```css
.my-form button {
  width: 100%;
  line-height: 1.5;
  padding: 15px 10px;
  border: 1px solid var(--borderFormEls);
  color: var(--white);
  background: var(--bgFormEls);
  transition: background-color 0.3s cubic-bezier(0.57, 0.21, 0.69, 1.25),
    transform 0.3s cubic-bezier(0.57, 0.21, 0.69, 1.25);
}

.my-form ::placeholder {
  color: inherit;
  /*Fix opacity issue on Firefox*/
  opacity: 1;
}

.my-form select:focus,
.my-form input:focus,
.my-form textarea:focus,
.my-form button:enabled:hover,
.my-form button:focus,
.my-form input[type="checkbox"]:focus+label {
  background: var(--bgFormElsFocus);
}

.my-form select:focus,
.my-form input:focus,
.my-form textarea:focus {
  transform: scale(1.02);
}

.my-form *:required,
.my-form select {
  background-repeat: no-repeat;
  background-position: center right 12px;
  background-size: 15px 15px;
}

.my-form *:required {
  background-image: url(https://s3-us-west-2.amazonaws.com/s.cdpn.io/162656/asterisk.svg);
}

.my-form select {
  background-image: url(https://s3-us-west-2.amazonaws.com/s.cdpn.io/162656/down.svg);
}

.my-form *:disabled {
  cursor: default;
```

```
            filter: blur(2px);
        }

        .my-form .required-msg {
            display: none;
            background: url(https://s3-us-west-2.amazonaws.com/s.cdpn.io/162656/asterisk.svg) no-repeat center left /
15px 15px;
            padding-left: 20px;
        }

        .my-form .btn-grid {
            position: relative;
            overflow: hidden;
            transition: filter 0.2s;
        }

        .my-form button {
            font-weight: bold;
        }

        .my-form button>* {
            display: inline-block;
            width: 100%;
            transition: transform 0.4s ease-in-out;
        }

        .my-form button .back {
            position: absolute;
            left: 50%;
            top: 50%;
            transform: translate(-110%, -50%);
        }

        .my-form button:enabled:hover .back,
        .my-form button:focus .back {
            transform: translate(-50%, -50%);
        }

        .my-form button:enabled:hover .front,
        .my-form button:focus .front {
            transform: translateX(110%);
        }
    </style>
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/css/bootstrap.min.css">
</head>

<body>
    <form class="my-form" action="/y_predict" method="post">
        <div class="container">
```

```
<h1>Enter Your Car Details to Predict Price</h1>
<ul>
  <li>
    <div class="row">
      <div class="grid grid-2 col-sm-6">
        <input id="regyear" type="text" name="regyear" required
          placeholder="Year Of Registration" />
      </div>
      <div class="grid grid-2 col-sm-6">
        <input id="regmonth" type="text" name="regmonth" required
          placeholder="Month Of Registration" />
      </div>
    </div>
  </li>
  <li>
    <div class="row">
      <div class="grid grid-2 col-sm-6">
        <input id="powerps" type="text" name="powerps" required placeholder="Power of Car in PS" />
      </div>
      <div class="grid grid-2 col-sm-6">
        <input id="kms" type="text" name="kms" required
          placeholder="Kilometers the Car has Driven" />
      </div>
    </div>
  </li>
  <li>
    <div class="row">
      <div class="grid grid-2 col-sm-6">
        <select id="gearbox" name="gearbox" required>
          <option selected disabled> Select Gear Box Type</option>
          <option value="manual">Manual</option>
          <option value="automatic">Automatic</option>
          <option value="not-declared">Not-Declared</option>
        </select>
      </div>
      <div class="grid grid-2 col-sm-6">
        <select id="dam" name="dam" required>
          <option selected disabled>Your Car is damaged or repaired</option>
          <option value="No">No</option>
          <option value="Yes">Yes</option>
          <option value="not-declared">Not-Declared</option>
        </select>
      </div>
    </div>
  </li>
  <li>
    <div class="row">
      <div class="grid grid-2 col-sm-6">
        <select id="modeltype" name="modeltype" required>
```

```
<option selected disabled>Model Type</option>
<option value="80">80</option>
<option value="90">90</option>
<option value="100">100</option>
<option value="145">145</option>
<option value="147">147</option>
<option value="156">156</option>
<option value="159">159</option>
<option value="200">200</option>
<option value="500">500</option>
<option value="850">850</option>
<option value="900">900</option>
<option value="911">911</option>
<option value="9000">9000</option>
<option value="1_reihe">1_Reihe</option>
<option value="1er">1Er</option>
<option value="2_reihe">2_Reihe</option>
<option value="3_reihe">3_Reihe</option>
<option value="300c">300C</option>
<option value="3er">3Er</option>
<option value="4_reihe">4_Reihe</option>
<option value="5_reihe">5_Reihe</option>
<option value="5er">5Er</option>
<option value="6_reihe">6_Reihe</option>
<option value="6er">6Er</option>
<option value="7er">7Er</option>
<option value="a_klasse">A_Klasse</option>
<option value="a1">A1</option>
<option value="a2">A2</option>
<option value="a3">A3</option>
<option value="a4">A4</option>
<option value="a5">A5</option>
<option value="a6">A6</option>
<option value="a8">A8</option>
<option value="accord">Accord</option>
<option value="agila">Agila</option>
<option value="alhambra">Alhambra</option>
<option value="almera">Almera</option>
<option value="altea">Altea</option>
<option value="amarok">Amarok</option>
<option value="andere">Andere</option>
<option value="antara">Antara</option>
<option value="arosa">Arosa</option>
<option value="astra">Astra</option>
<option value="auris">Auris</option>
<option value="avensis">Avensis</option>
<option value="aveo">Aveo</option>
<option value="aygo">Aygo</option>
<option value="b_klasse">B_Klasse</option>
```

```
<option value="b_max">B_Max</option>
<option value="beetle">Beetle</option>
<option value="berlingo">Berlingo</option>
<option value="bora">Bora</option>
<option value="boxster">Boxster</option>
<option value="bravo">Bravo</option>
<option value="c_klasse">C_Klasse</option>
<option value="c_max">C_Max</option>
<option value="c_reihe">C_Reihe</option>
<option value="c1">C1</option>
<option value="c2">C2</option>
<option value="c3">C3</option>
<option value="c4">C4</option>
<option value="c5">C5</option>
<option value="caddy">Caddy</option>
<option value="calibra">Calibra</option>
<option value="captiva">Captiva</option>
<option value="carisma">Carisma</option>
<option value="carnival">Carnival</option>
<option value="cayenne">Cayenne</option>
<option value="cc">Cc</option>
<option value="ceed">Ceed</option>
<option value="charade">Charade</option>
<option value="cherokee">Cherokee</option>
<option value="citigo">Citigo</option>
<option value="civic">Civic</option>
<option value="cl">Cl</option>
<option value="clio">Clio</option>
<option value="clk">Clk</option>
<option value="clubman">Clubman</option>
<option value="colt">Colt</option>
<option value="combo">Combo</option>
<option value="cooper">Cooper</option>
<option value="cordoba">Cordoba</option>
<option value="corolla">Corolla</option>
<option value="corsa">Corsa</option>
<option value="cr_reihe">Cr_Reihe</option>
<option value="croma">Croma</option>
<option value="crossfire">Crossfire</option>
<option value="cuore">Cuore</option>
<option value="cx_reihe">Cx_Reihe</option>
<option value="defender">Defender</option>
<option value="delta">Delta</option>
<option value="discovery">Discovery</option>
<option value="discovery_sport">Discovery_Sport</option>
<option value="doblo">Doblo</option>
<option value="ducato">Ducato</option>
<option value="duster">Duster</option>
<option value="e_klasse">E_Klasse</option>
```

```
<option value="elefantino">Elefantino</option>
<option value="eos">Eos</option>
<option value="escort">Escort</option>
<option value="espace">Espace</option>
<option value="exeo">Exeo</option>
<option value="fabia">Fabia</option>
<option value="fiesta">Fiesta</option>
<option value="focus">Focus</option>
<option value="forester">Forester</option>
<option value="forfour">Forfour</option>
<option value="fortwo">Fortwo</option>
<option value="fox">Fox</option>
<option value="freelander">Freelander</option>
<option value="fusion">Fusion</option>
<option value="g_klasse">G_Klasse</option>
<option value="galant">Galant</option>
<option value="galaxy">Galaxy</option>
<option value="getz">Getz</option>
<option value="gl">Gl</option>
<option value="glk">Glk</option>
<option value="golf">Golf</option>
<option value="grand">Grand</option>
<option value="i_reihe">I_Reihe</option>
<option value="i3">I3</option>
<option value="ibiza">Ibiza</option>
<option value="impreza">Impreza</option>
<option value="insignia">Insignia</option>
<option value="jazz">Jazz</option>
<option value="jetta">Jetta</option>
<option value="jimny">Jimny</option>
<option value="juke">Juke</option>
<option value="justy">Justy</option>
<option value="ka">Ka</option>
<option value="kadett">Kadett</option>
<option value="kaefer">Kaefer</option>
<option value="kalina">Kalina</option>
<option value="kalos">Kalos</option>
<option value="kangoo">Kangoo</option>
<option value="kappa">Kappa</option>
<option value="kuga">Kuga</option>
<option value="laguna">Laguna</option>
<option value="lancer">Lancer</option>
<option value="lanos">Lanos</option>
<option value="legacy">Legacy</option>
<option value="leon">Leon</option>
<option value="lodgy">Lodgy</option>
<option value="logan">Logan</option>
<option value="lupo">Lupo</option>
<option value="lybra">Lybra</option>
```

```
<option value="m_klasse">M_Klasse</option>
<option value="m_reihe">M_Reihe</option>
<option value="materia">Materia</option>
<option value="matiz">Matiz</option>
<option value="megane">Megane</option>
<option value="meriva">Meriva</option>
<option value="micra">Micra</option>
<option value="mii">Mii</option>
<option value="modus">Modus</option>
<option value="mondeo">Mondeo</option>
<option value="move">Move</option>
<option value="musa">Musa</option>
<option value="mustang">Mustang</option>
<option value="mx_reihe">Mx_Reihe</option>
<option value="navara">Navara</option>
<option value="niva">Niva</option>
<option value="not-declared">Not-Declared</option>
<option value="note">Note</option>
<option value="nubira">Nubira</option>
<option value="octavia">Octavia</option>
<option value="omega">Omega</option>
<option value="one">One</option>
<option value="outlander">Outlander</option>
<option value="pajero">Pajero</option>
<option value="panda">Panda</option>
<option value="passat">Passat</option>
<option value="phaeton">Phaeton</option>
<option value="picanto">Picanto</option>
<option value="polo">Polo</option>
<option value="primera">Primera</option>
<option value="ptcruiser">Ptcruiser</option>
<option value="punto">Punto</option>
<option value="q3">Q3</option>
<option value="q5">Q5</option>
<option value="q7">Q7</option>
<option value="qashqai">Qashqai</option>
<option value="r19">R19</option>
<option value="range_rover">Range_Rover</option>
<option value="range_rover_evoque">Range_Rover_Evoque</option>
<option value="range_rover_sport">Range_Rover_Sport</option>
<option value="rangerover">Rangerover</option>
<option value="rav">Rav</option>
<option value="rio">Rio</option>
<option value="roadster">Roadster</option>
<option value="roomster">Roomster</option>
<option value="rx_reihe">Rx_Reihe</option>
<option value="s_klasse">S_Klasse</option>
<option value="s_max">S_Max</option>
<option value="s_type">S_Type</option>
```

```
<option value="s60">S60</option>
<option value="samara">Samara</option>
<option value="sandero">Sandero</option>
<option value="santa">Santa</option>
<option value="scenic">Scenic</option>
<option value="scirocco">Scirocco</option>
<option value="seicento">Seicento</option>
<option value="serie_2">Serie_2</option>
<option value="serie_3">Serie_3</option>
<option value="sharan">Sharan</option>
<option value="signum">Signum</option>
<option value="sirion">Sirion</option>
<option value="sl">Sl</option>
<option value="slk">Slk</option>
<option value="sorento">Sorento</option>
<option value="spark">Spark</option>
<option value="spider">Spider</option>
<option value="sportage">Sportage</option>
<option value="sprinter">Sprinter</option>
<option value="stilo">Stilo</option>
<option value="superb">Superb</option>
<option value="swift">Swift</option>
<option value="terios">Terios</option>
<option value="tigra">Tigra</option>
<option value="tiguan">Tiguan</option>
<option value="toledo">Toledo</option>
<option value="touareg">Touareg</option>
<option value="touran">Touran</option>
<option value="transit">Transit</option>
<option value="transporter">Transporter</option>
<option value="tt">Tt</option>
<option value="tucson">Tucson</option>
<option value="twingo">Twingo</option>
<option value="up">Up</option>
<option value="v_klasse">V_Klasse</option>
<option value="v40">V40</option>
<option value="v50">V50</option>
<option value="v60">V60</option>
<option value="v70">V70</option>
<option value="vectra">Vectra</option>
<option value="verso">Verso</option>
<option value="viano">Viano</option>
<option value="vito">Vito</option>
<option value="vivaro">Vivaro</option>
<option value="voyager">Voyager</option>
<option value="wrangler">Wrangler</option>
<option value="x_reihe">X_Reihe</option>
<option value="x_trail">X_Trail</option>
<option value="x_type">X_Type</option>
```

```
            <option value="xc_reihe">Xc_Reihe</option>
            <option value="yaris">Yaris</option>
            <option value="yeti">Yeti</option>
            <option value="ypsilon">Ypsilon</option>
            <option value="z_reihe">Z_Reihe</option>
            <option value="zafira">Zafira</option>

        </select>
    </div>
    <div class="grid grid-2 col-sm-6">
        <select id="brand" class="input" name="brand" required>
            <option selected disabled>Brand of the Car</option>
            <option value="audi">Audi</option>
            <option value="alfa_romeo">Alfa_Romeo</option>
            <option value="fiat">Fiat</option>
            <option value="volvo">Volvo</option>
            <option value="saab">Saab</option>
            <option value="porsche">Porsche</option>
            <option value="mazda">Mazda</option>
            <option value="peugeot">Peugeot</option>
            <option value="bmw">Bmw</option>
            <option value="chrysler">Chrysler</option>
            <option value="mercedes_benz">Mercedes_Benz</option>
            <option value="honda">Honda</option>
            <option value="opel">Opel</option>
            <option value="seat">Seat</option>
            <option value="nissan">Nissan</option>
            <option value="volkswagen">Volkswagen</option>
            <option value="hyundai">Hyundai</option>
            <option value="mitsubishi">Mitsubishi</option>
            <option value="lancia">Lancia</option>
            <option value="toyota">Toyota</option>
            <option value="renault">Renault</option>
            <option value="chevrolet">Chevrolet</option>
            <option value="suzuki">Suzuki</option>
            <option value="kia">Kia</option>
            <option value="citroen">Citroen</option>
            <option value="ford">Ford</option>
            <option value="jaguar">Jaguar</option>
            <option value="skoda">Skoda</option>
            <option value="rover">Rover</option>
            <option value="smart">Smart</option>
            <option value="mini">Mini</option>
            <option value="lada">Lada</option>
            <option value="daewoo">Daewoo</option>
            <option value="jeep">Jeep</option>
            <option value="daihatsu">Daihatsu</option>
            <option value="subaru">Subaru</option>
            <option value="dacia">Dacia</option>
```

```
                        <option value="land_rover">Land_Rover</option>
                        <option value="trabant">Trabant</option>
                        <option value="sonstige_autos">Sonstige_Autos</option>
                    </select>
                </div>
            </div>
        </li>
        <li>
            <div class="row">
                <div class="grid grid-2 col-sm-6">
                    <select id="fuel" class="input" name="fuel" required>
                        <option selected disabled>Fuel type of the car</option>
                        <option value="petrol">Petrol</option>
                        <option value="diesel">Diesel</option>
                        <option value="lpg">Lpg</option>
                        <option value="hybrid">Hybrid</option>
                        <option value="cng">Cng</option>
                        <option value="electric">Electric</option>
                        <option value="not-declared">Not-Declared</option>
                        <option value="others">Others</option>
                    </select>
                </div>
                <div class="grid grid-2 col-sm-6">
                    <select id="vehicletype" class="input" name="vehicletype" required>
                        <option selected disabled>Vehicle Type</option>
                        <option value="convertible">Convertible</option>
                        <option value="limousine">Limousine</option>
                        <option value="combination">Combination</option>
                        <option value="coupe">Coupe</option>
                        <option value="samll car">Samll Car</option>
                        <option value="bus">Bus</option>
                        <option value="not-declared">Not-Declared</option>
                        <option value="others">Others</option>
                    </select>
                </div>
            </div>
        </li>
        <li>
            <button class="btn-grid" type="submit">Predict The Price</button>
        </li>
    </ul>
</div>
</form>
</body>

</html>
```

resalepredict.html

```
<!DOCTYPE html>
<html>

<head>
   <title> CAR RESALE VALUE PREDICTION</title>
   <style>
      div {
         padding: 8px;
      }

      h1,h2 {
         text-align: center;
         text-transform: uppercase;
         color: #273342;
      }

      p {
         text-indent: 50px;
         text-align: justify;
         letter-spacing: 3px;
      }

      a {
         text-decoration: underline;
         color: #543fd7;
      }
   </style>
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
</head>

<body>
   <div>
      <h1>The resale value predicted is </h1>
      <h2><i class="fa fa-inr"></i>{{ypred}}</h2>
   </div>
</body>

</html>
```

13.2 GITHUB & PROJECT DEMO LINK

13.2.1 GITHUB

https://github.com/IBM-EPBL/IBM-Project-19633-1659702729

13.2.2 PROJECT DEMO LINK

https://drive.google.com/file/d/19BV2YciQJsHpzEhLnkcdwqnMDuaMPWPX/view?usp=sharing