CAR RESALE VALUE PREDICTION

IBM PROJECT

Submitted by

SAI MOUNISH P	**142219104105**
ROACH AMALAN S	**142219104101**
RANJITH K	**142219104096**
ROSHAN R	**142219104103**

Team ID : PNT2022TMID21699

in a partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

In

COMPUTER SCIENCE AND ENGINEERING

SRM VALLIAMMAI ENGINEERING COLLEGE
(AN AUTONOMOUS INSTITUTION)
SRM NAGAR, KATTANKULATHUR,
ANNA UNIVERSITY: CHENNAI 600 025

NOVEMBER 2022

1. INTRODUCTION

1.1 Project Overview

Determining whether the listed price of a used car is a challenging task, due to the many factors that drive a used vehicle's price on the market. The focus of this project is developing machine learning models that can accurately predict the price of a used car based on its features, in order to make informed purchases. We implement and evaluate various learning methods on a dataset consisting of the sale prices of different makes and models across cities in the United States. Our results show that Random Forest model and K-Means clustering with linear regression yield the best results, but are compute heavy. Conventional linear regression also yielded satisfactory results, with the advantage of a significantly lower training time in comparison to the aforementioned methods

1.2 Purpose

Deciding whether a used car is worth the posted price when you see listings online can be difficult. Several factors, including mileage, make, model, year, etc. can influence the actual worth of a car. From the perspective of a seller, it is also a dilemma to price a used car appropriately[2-3]. Based on existing data, the aim is to use machine learning algorithms to develop models for predicting used car prices.

2. LITERATURE SURVEY

SI No.	TITLE	JOURNAL	AUTHOR	CHALLENGES/ FUTURE SCOPE
1.	Used car price prediction	IRJET	praful rana, deep pandiya, dhawal kotak	n future this machine learning model may bind with various website which can provide real time data for price prediction. Also we may add large historical data of car price which can help to improve accuracy of the machine learning model. We can build an android app as user interface for interacting with user. For better performance, we plan to judiciously design deep learning network structures, use adaptive learning rates and train

				on clusters of data rather than the whole dataset.
2.	used car price prediction and life span	IARJSET	aditya nikhade , rohan borde	This Project In machine learning model that will be connected with may dataset and with various website which can provide real time data for price prediction Will Stored in their site or GitHub. Also, we may add big amount of data of car price which can help an improve accuracy of the machine learning model . We also trying to develop an android app as user interface for interacting and user friendly with user. For better performance of the model, we also plan a to use neural network.

3.	vehicle resale price prediction using machine learning	Juni Khyat (UGC Care Group I Listed urnal)	B.Lavanya, Sk.Reshma, N.Nikitha, M.Namitha, L.Kanya Kumar, S.Kishore Babu,	In this paper, four distinctive AI procedures have been utilized to figure the cost of pre-owned vehicles in Mauritius. The mean blunder with direct relapse was about Rs 51,000 while for kNN it was about Rs 27,000 for Nissan vehicles and about Rs 45,000 for Toyota vehicles. J48 and Naïve Bayes exactness hung between 60-70% for various blends of boundaries. The primary shortcoming of choice trees and credulous bayes is their powerlessness to deal with yield classes with numeric qualities. Consequently, the value quality must be ordered into classes which contained a

				scope of costs yet this clearly presented further justification for errors. The primary limit of this examination is the low number of records that have been utilized. As future work, we plan to gather more information and to utilizes further developed methods like counterfeit neural organizations, fluffy logic and hereditary calculations to foresee vehicle costs.
4.	Predicting Used Car	CS 229 Project Report	Kshitij Kumbar,	For better performance, we plan

	Prices		Pranav Gadre and Varun Nayak	to judiciously design deep learning network structures, use adaptive learning rates and train on clusters of data rather than the whole dataset. To correct for overfitting in Random Forest, different selections of features and number of trees will be tested to check for change in performance.

5.	Used Cars Price Prediction using Supervised Learning Techniques	International Journal of Engineering and Advanced Technology	Mukkesh Ganesh	The prediction error rate of all the models was well under the accepted 5% of error. But, on further analysis, the mean error of the regression tree model was found to be more than the mean error rate of the multiple regression and lasso regression models. Even though for some seeds the regression tree has better accuracy, its error rates are higher for the rest. This has been confirmed by performing an ANOVA. Also, the post-hoc test revealed that the error rates in multiple regression models and lasso regression models aren't significantly different from each

				other. To get even more accurate models, we can also choose more advanced machine learning algorithms such as random forests, an ensemble learning algorithm which creates multiple decision/regression trees, which brings down overfitting massively or Boosting, which tries to bias the overall model by weighing in the favor of good performers. More data from newer websites and different countries can also be scraped and this data can be used to retrain these models to check for reproducibility.

6.	predictive analysis of used car prices using machine learning	International Research Journal of Modernization in Engineering Technology and Science	Ashutosh Datt Sharma ,Vibhor Sharma,Sahil Mittal,Gautam Jain,Sudha Narang	Predicting prices of a used car is a challenging task because of a high number of features and parameters that should be considered to generate accurate results. The first and foremost step is data gathering and preprocessing data. Then a model was defined and created for implementing algorithms and generating results. After applying various regression algorithms on the model, it could be concluded that Decision Tree Algorithm was the best performer with highest r2 score of 0.95 which simply signified the fact that it generated the most accurate

				predictions as reflected by the Original v/s Prediction line graph. Apart from a best r2 score, Decision Tree also had the least Mean Squared Error and Root Mean Squared Values that shows that the errors in predictions were least among all and therefore the results generated are highly accurate. .

7.	Price Prediction for Used Cars	Mid Sweden University.	Marcus Collard	the best potential for development of a consumer tool for evaluating used cars or a particular subset of used cars. The results show that Random Forest Regression performed the best on all performance metrics and for all price percentile subsets of used cars. It was also much better able to approximate the depreciation.
8.	Car Price Prediction using Machine Learning Techniques	TEM Journal. Volume 8	Enis Gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric	Car price prediction can be a challenging task due to the high number of attributes that should be considered for the accurate prediction. The major step in the prediction process is collection and preprocessing of the data. In this research,

				PHP scripts were built to normalize, standardize and clean data to avoid unnecessary noise for machine learning algorithms.
9.	Used Cars Price Prediction and Valuation using Data Mining Techniques	Rochester Institute of Technology	Abdulla AlShared	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. As a result of preprocessing and transformation, Random Forest Regress or came out on top with 95% accuracy followed by Bagging Regress or with 88%. Each experiment was performed in realtime within the Google environment. In comparison to the system's integrated Jupiter notebook and Anaconda's platform, algorithms took less training time in Google .
10.	Consumer preferences for electric vehicles: a Consumer preferences for electric vehicles: a	Transport Reviews	Fanchao Liao, Eric Molin , Bert van Wee	In general, the effect of individualspecific variables on EV preference remains an open question. Psychological variables are the exception and have a

				proven stable effect, shown by several studies. For socioeconomic and demographic variables,the impact is unclear and sensitive to small changes in model specification. The direction of the effect is also ambiguous since existing evidenceis contradictory. Other variables are only included in a few studies, therefore their effects are as yet inconclusive. In most cases, the correlation between all these variables has not been controlled for to avoid self - selection bias.More research is definitely necessary to clarify these currently fuzzy relationships and other methods are needed to add more and confidence to theresults

2.1 Existing Problem

The real reason that this problem exist is in this car resale value prediction system cant predict exact price as brand owners price. This just predicts approx. the value by interior and exterior, bs4 and bs6, petrol or diesel.

2.2 References

[1] NATIONAL TRANSPORT AUTHORITY. 2014. Available from: http://nta.gov.mu/English/Statistics/Pages/Archive.aspx [Accessed 15 January 2014].

[2] MOTORS MEGA. 2014. Available from: http://motors.mega.mu/news/2013/12/17/auto-market-8-decrease-sales-newcars/ [Accessed 17 January 2014].

[3] LISTIANI, M., 2009. Support Vector Regression Analysis for Price Prediction in a Car Leasing Application. Thesis (MSc). Hamburg University of Technology.

[4] RICHARDSON, M., 2009. Determinants of Used Car Resale Value. Thesis (BSc). The Colorado College.

[5] WU, J. D., HSU, C. C. AND CHEN, H. C., 2009. An expert system of price forecasting for used cars using adaptive neuro-fuzzy inference. Expert Systems with Applications. Vol. 36, Issue 4, pp. 7809-7817.

[6] DU, J., XIE, L. AND SCHROEDER S., 2009. Practice Prize Paper - PIN Optimal Distribution of Auction Vehicles System: Applying Price Forecasting, Elasticity Estimation and Genetic Algorithms to Used-Vehicle Distribution. Marketing Science, Vol. 28, Issue 4, pp. 637-644.

[7] GONGGI, S., 2011. New model for residual value prediction of used cars based on BP neural network and non-linear curve fit. In: Proceedings of the 3 rd IEEE International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), Vol 2. pp. 682-685, IEEE Computer Society, Washington DC, USA.

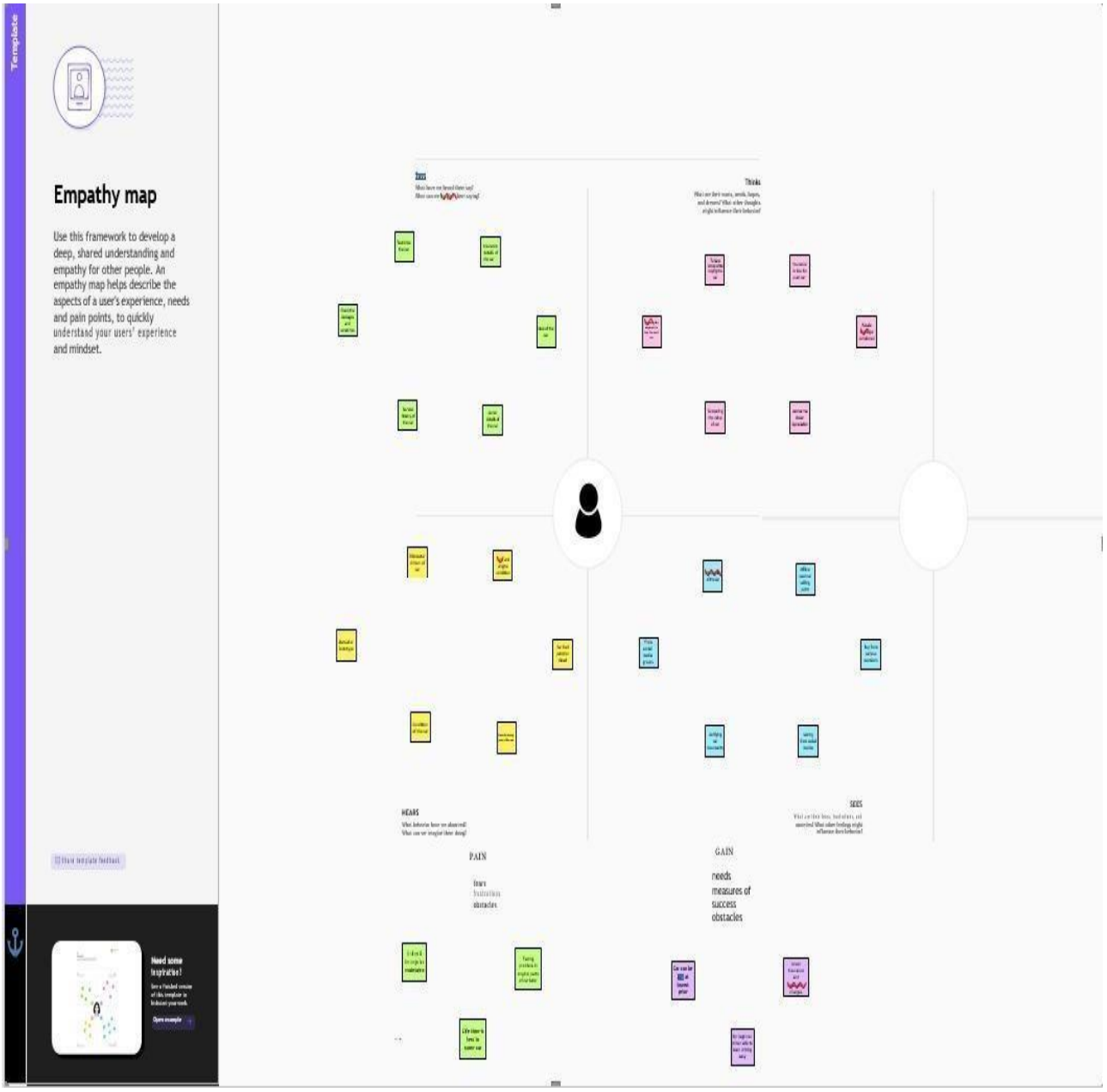
[8] LEXPRESS.MU ONLINE. 2014. Available from: http://www.lexpress.mu/ [Accessed 17 January 2014]

[9] LE DEFI MEDIA GROUP. 2014. Available from: http://www.defimedia.info/ [Accessed 17 January 2014]

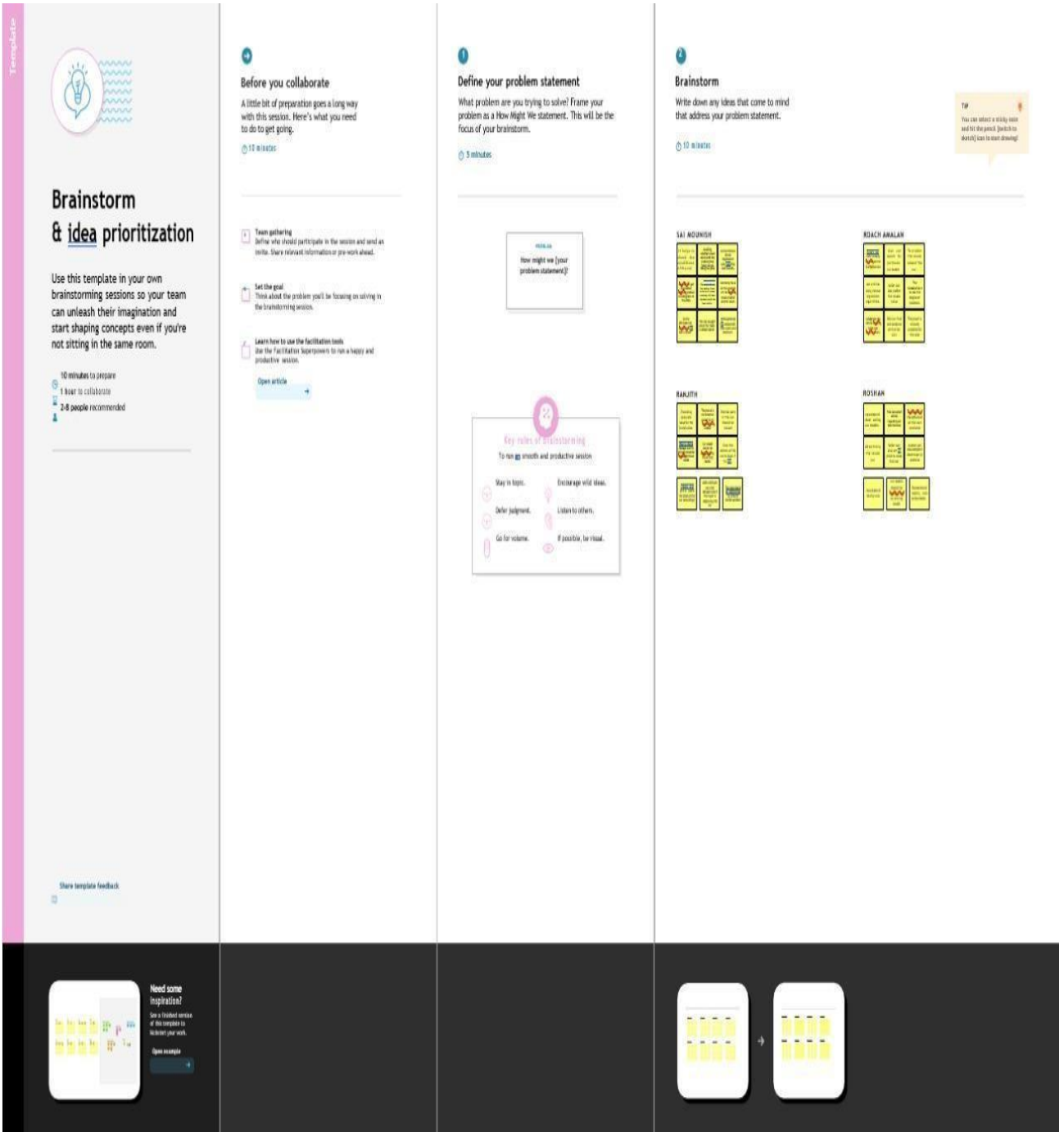
[10] GELMAN, A. AND HILL, J., 2006. Data Analysis Using Regression and Multilevel Hierarchical Models. Cambridge University Press, New York, USA.

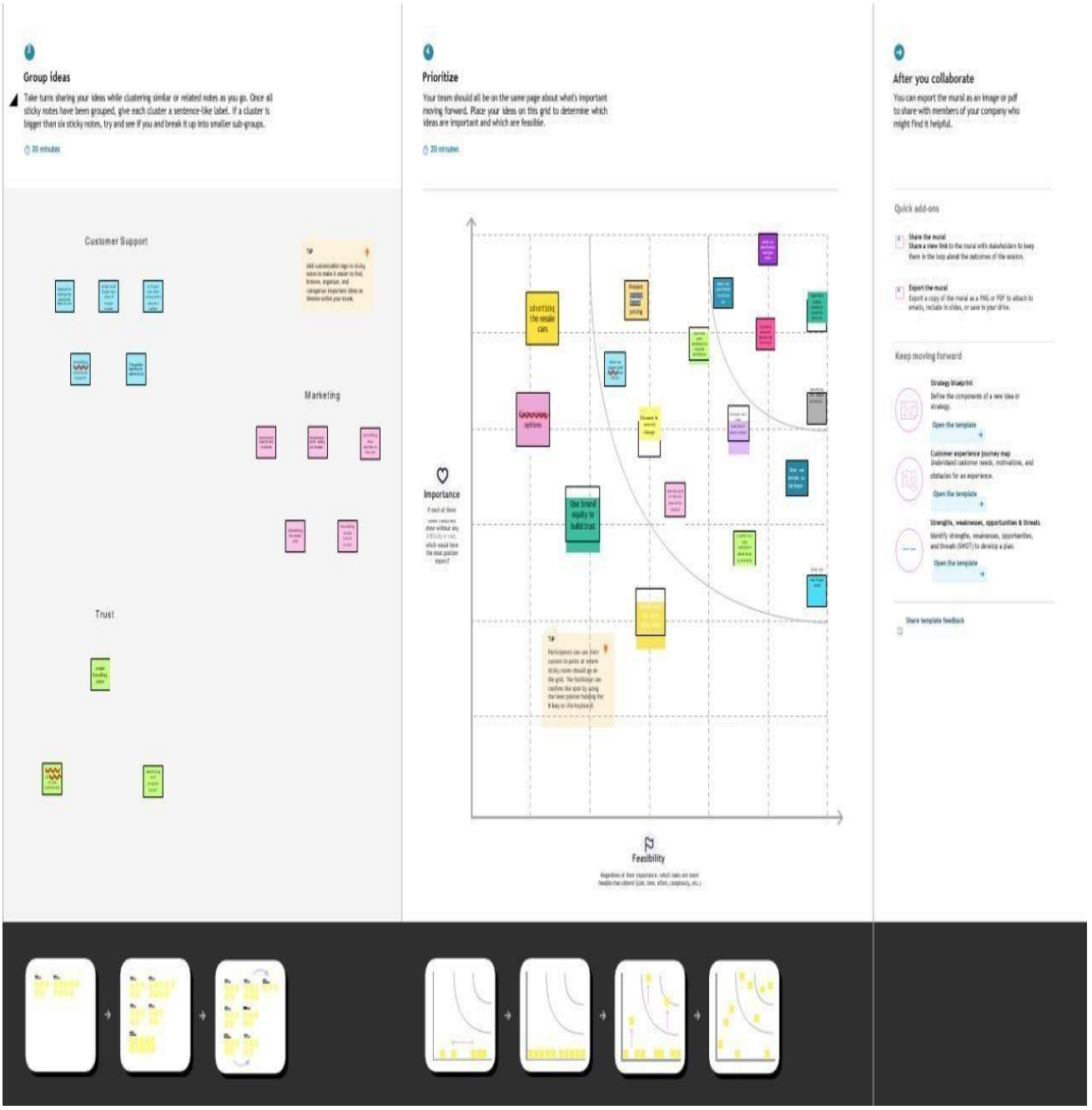
2.3 Problem Statement Definition

3.1 Empathy Map Canvas

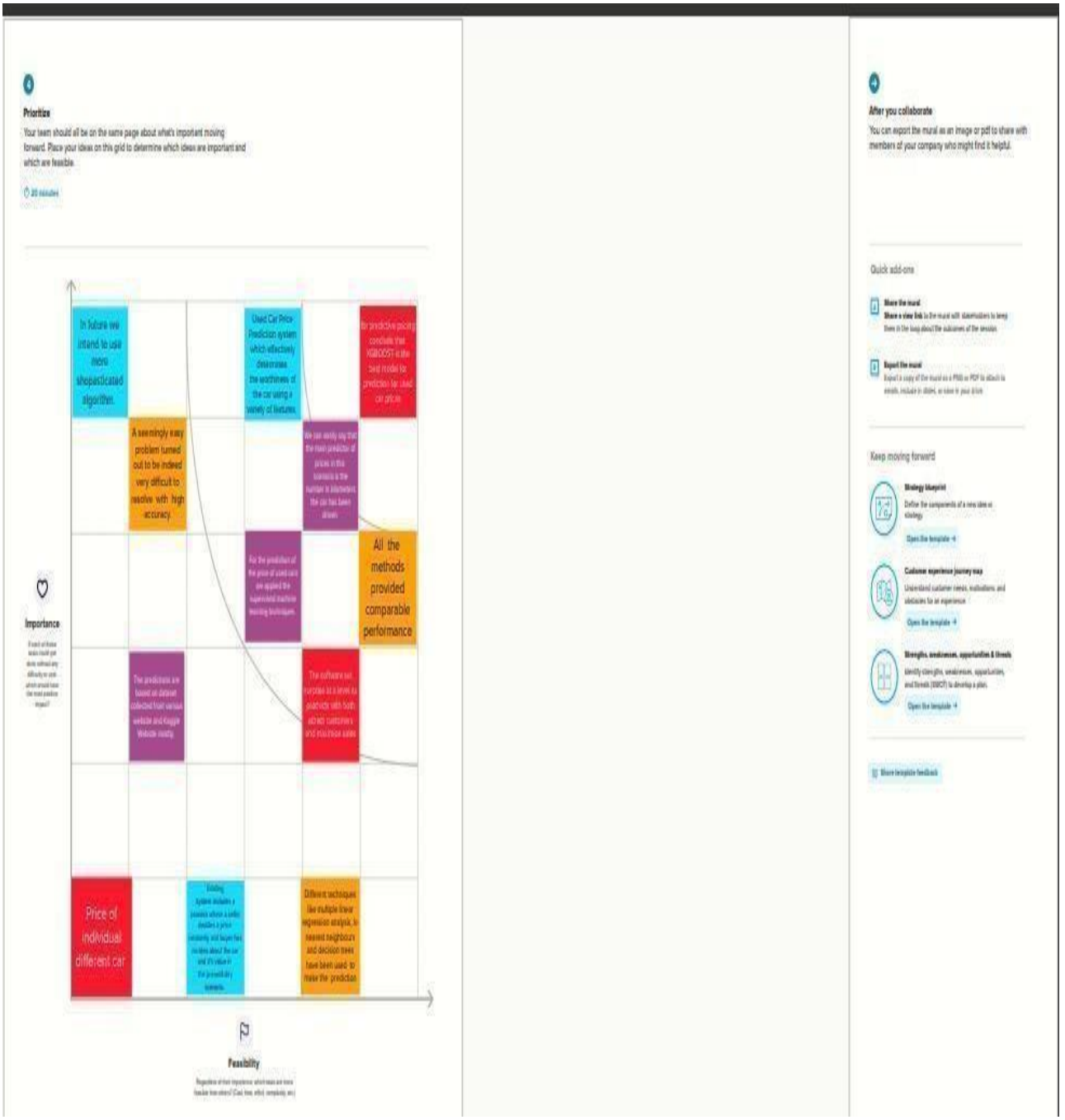


3.2 Ideation & Brainstorming




Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.
20 minutes
Customer Support
Marketing
Trust
Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.
20 minutes
advertising the resale cars
Our brand equity to build trust
Importance
Feasibility
After you collaborate
You can export the mural as an image or pdf to share with members of your company who might find it helpful.
Quick add-ons
Share the mural
Export the mural
Keep moving forward
Strategy blueprint
Define the components of a new idea or strategy.
Open the template
Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template
Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template
Share template feedback

3.3 Proposed Solution



S.No.	Parameter	Decription
1.	Problem Statement (Problem to be solved)	The main aim of this project is to predict the price of used cars using the various Machine Learning (ML) models. The project should take parameters related to used car as inputs and enable the customers to make decisions by their own
2.	Feasibility of Idea	New cars of a particular make, model, and year all have the same retail price, excluding optional features. This price is set by the manufacturer. Used car, however, are subject to supply-and-demand pricing. Further, used cars have additional attributes that factor into the price. These include the condition, milage, and repair history, which sets cars that may have shared a retail price apart.
3.	Novelty	Used car price prediction is effectively used to determine the worthiness of the car by their own within few minutes by using various features such as year, model, mileage(km), etc
4.	Social Impact/ Customer Satisfaction	If the user wants to buy or sell a own car it helps users to predict the correct valuation by their own. A loss function is to be optimized and mainly a weak learner can make predictions for used cars easily.

5.	Business Model(Revenue Model)	It helps users to predict the correct valuation of the car remotely with perfect valuation and without human intervention like car dealers in the process to eliminate biased valuation predicted by the dealer. Using Stored data and dataset provided.
6.	Scalability of the Solution	Which of the models and parameters gives the best overall accuracy in making price predictions for used cars. The optimal parameters were determined in the process of implementing the models, and thus each model was implemented with the parameters that yielded the best performance by trial and error. All of the models approximated geometric appreciation, meaning that a constant percentage of value is lost every year independent of the age of the vehicle.Random Forest Regression had a significantly higher assessed average depreciation at approximately 13.8%, compared to the others with 9.7%. This is closer to the range of 15%-31% assessed by Karl Storchmann in his analysis of international depreciation rates

3.4 Proposed Solution Fit

4. REQUIREMENT ANALYSIS

4.1 Functional Requirements

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	Core functionality	Recognize the human handwritten digits from different sources like images, papers, touch screens, etc, and classify them into 10 predefined classes (0-9)
FR-3	Access	Able to copy the recognised digits, Focus a part of the image manually.
FR-4	Network	The database has to be updated for training for more accuracy.

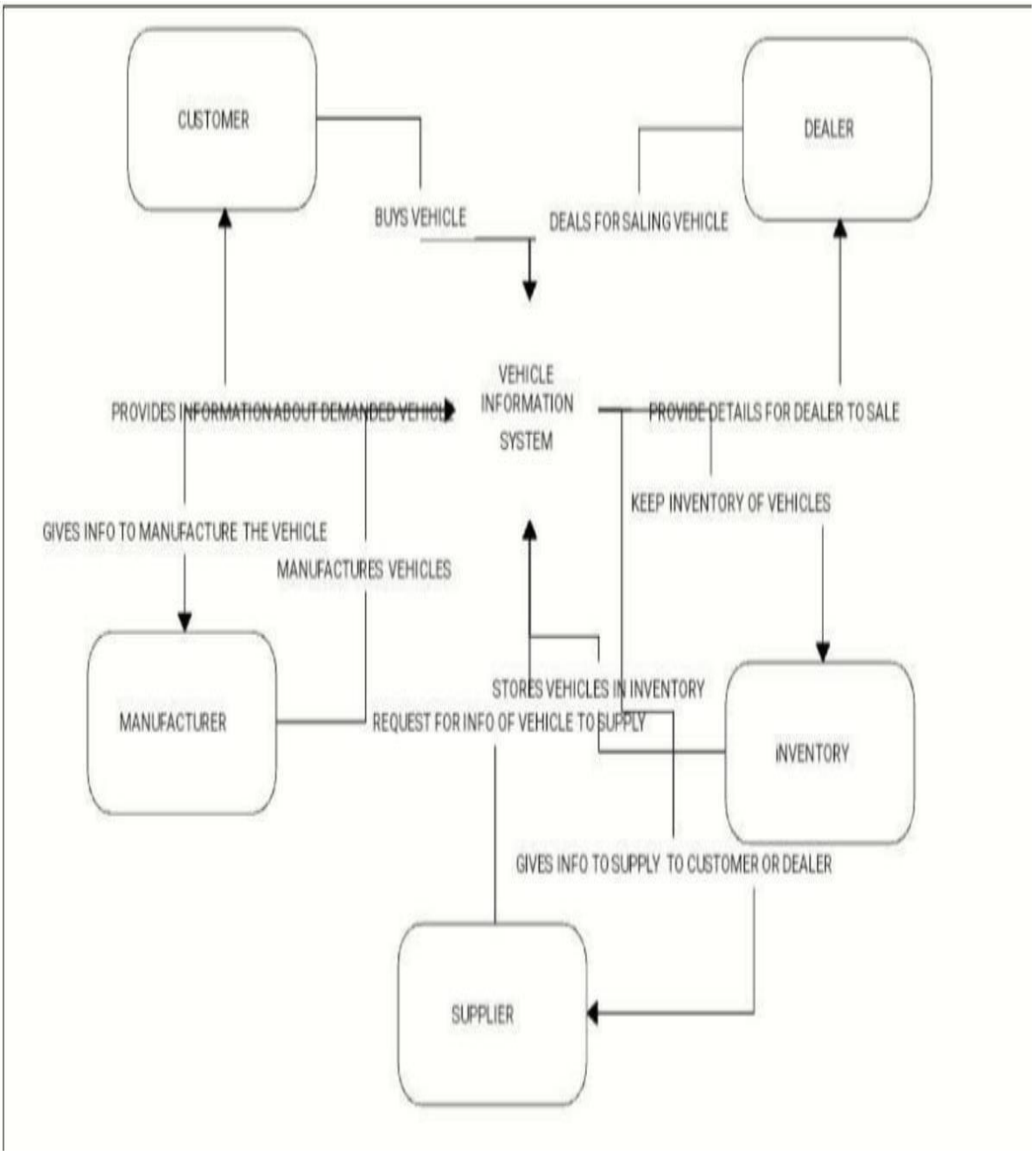
4.2 Non-Functional Requirements

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	Recognising handwritten information such as reading postal addresses, bank check amounts, and forms.
NFR-2	Security	When the image is passed to recognise a particular area of digit(s), the image will not be stored at the backend.
NFR-3	Reliability	CNN has shown remarkable abilities in offline handwritten character recognition of Arabic language; handwritten Tamil character recognition; Telugu character recognition, handwritten Urdu text recognition, handwritten character recognition in Indic

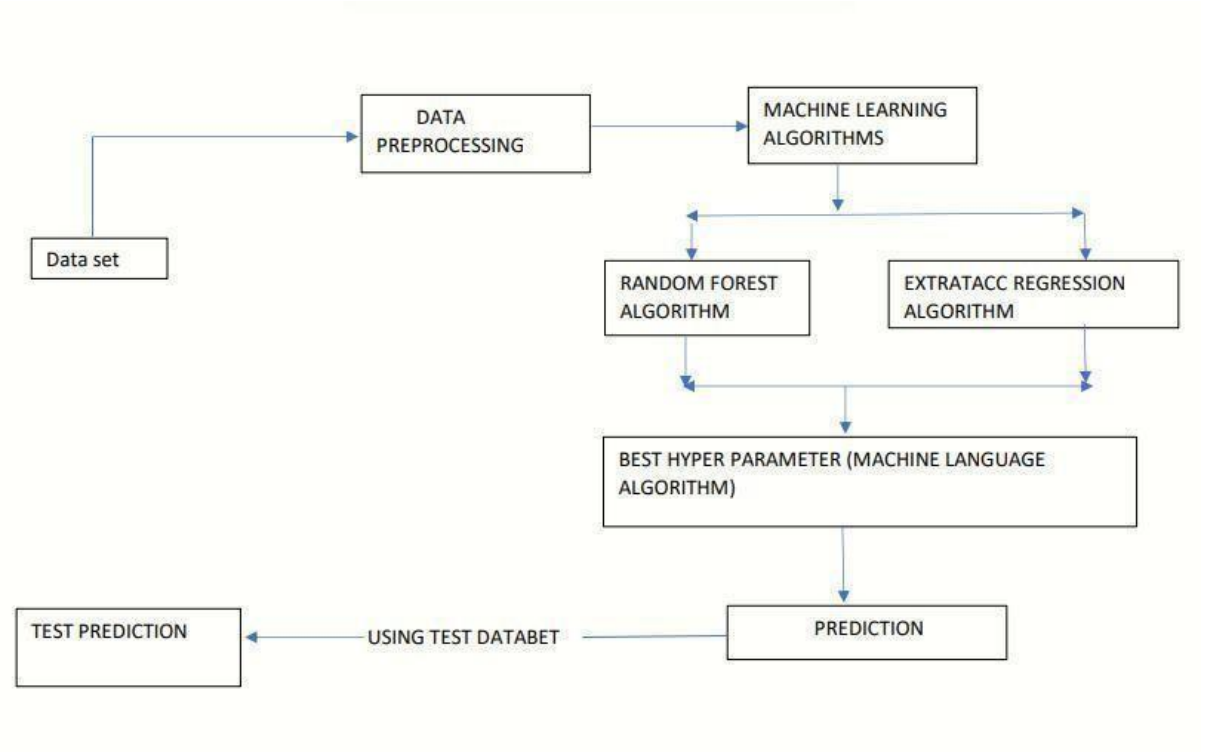
		scripts [44] and Chinese handwritten text recognition.
NFR-4	Performance	Hyper-parameters are, namely, activation function, number of epochs, kernel size, learning rate, hidden units, hidden layers, etc. that are responsible for the performance of the system.
NFR-5	Availability	There is no maintenance time separately for the servers to be down or can be accessed offline also.
NFR-6	Scalability	System will be such that it is easy to change, update, or add features later on.

5. PROJECT DESIGN

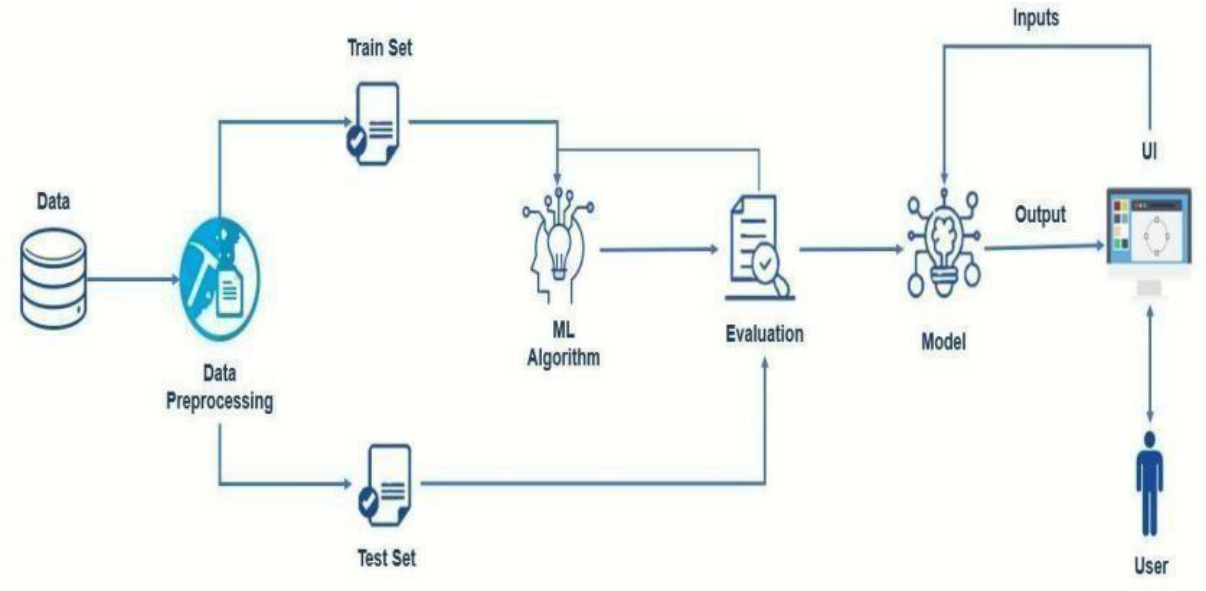
5.1 Data Flow Diagram



5.2 Solution Architecture



Technical Architecture:



5.3 User Stories:

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			registration fee, insurance cost, repair work and general upkeep.			
	Create a Target List	USN-2	Once you have agreed on a budget, start making a list of requirements for your vehicle. You must also choose the type of vehicle you want. You can choose from SUVs, sedans, small cars and electric vehicles. It is recommended to check the reviews and ratings of the car you plan to purchase.		High	Sprint-1
	Research Your Options	USN-3	Used car dealerships are now presenting almost every corner of the city, everywhere in India. You can find the best dealer in town either by word of mouth or by comparing dealers online. Finding good dealers online is a fairly simple process. Just shortlist some popular second-hand car dealers and compare options available, cost, service and customer reviews before choosing the one for yourself.		Medium	Sprint-2
	Check the Vehicle's History	USN-4	Once you have explored various options and have narrowed down your search list, it is time to check the vehicle's health report. Check what kind of maintenance or repair works has it undergone. Double-check if the vehicle has ever been involved in a collision. If you are buying a used car in India, it is advisable to avoid buying a car that has been involved in an accident.		High	Sprint-1
	Call the Seller	USN-5	Contact the seller to double-check the information you have gathered about the vehicle. If you are buying from an individual seller, find out why they are selling the car and if there are any mechanical concerns. If you are considering a dealer, call to check the availability of the car. If everything goes fine, book an appointment for a test drive.		High	Sprint-1

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
	Test Drive	USN-6	A test drive will give you a clear idea about your shortlisted used car's condition. Take the car for a drive on different types of roads and cover a distance of at least four to five kilometers. You must also check the condition of the brakes and clutch while driving. Ensure that the speedometer and the distance recorder are working properly. If there is a vibration in the steering, it could mean some major issues with the engine.		High	
	Get a Professional to Inspect the Car	USN-7	When buying a used car, get a professional mechanic to inspect the car before you pay for it. If you buy a used car from a reputable dealer, the chances of receiving a damaged model are slim. Buying from a private seller, on the other hand, may necessitate a complete inspection by a skilled mechanic.		Medium	
	Double Check the Vehicle's Papers	USN-8	Before finalising the used car, it is advisable to check the papers properly. Check for the car's registration certificate; match the vehicle's engine number and chassis number. Check the insurance paper, PUC certificate along with the original sales invoice. This way, you can make sure the car you are buying is not stolen from its previous owner.		High	
	Negotiate Well	USN-9	This is when the real fun begins. Since you would have already set a budget for the car purchase, stick to it and negotiate with the seller over anything you deem important such as a major dent or bad paintwork. Since the cost of a used car is the seller's decision, make sure to negotiate well.		High	
	Used Car Finance	USN-10	Today, many financial institutions offer a loan for the purchase of used cars. If you are under a budget constraint, you may avail of this option. Before applying for a loan, compare the used car finance rates with different		Medium	

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
			lenders and check your used car loan eligibility with the lender of your choice. If you have a good profile and strong creditworthiness, you may seal a better deal on used car finance			
	Ownership Transfer	USN-11	The ownership of a car is transferred with its sale. The previous owner of the car must inform about the transfer to the RTO under which the vehicle is originally registered. This process must be initiated within 14 days along with a letter of intent and the details of the new owner.	I can access my account / dashboard	High	
Straight away	Drive Away	USN-12	Once you are done with the above formalities, it is time to announce your purchase and be a proud car owner. You can now spin off the car to your home or wherever the road calls you		High	

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Home Page	USN-1	As a user, I can view the home page of the web application.	20	Low	Sai Mounish P
Sprint-2	Car resale value display	USN-2	As a user, I can be redirected to the data entry page	20	Medium	Roach Amalan
Sprint-3	Data Entry	USN-3	As a user, I can enter my car details in the re4quired fields.	20	Medium	Ranjith K
Sprint-4	Resale Value Prediction	USN-4	As a user, I expect the application to predict the resale value of my car.	20	Medium	Roshan 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	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

VELOCITY:

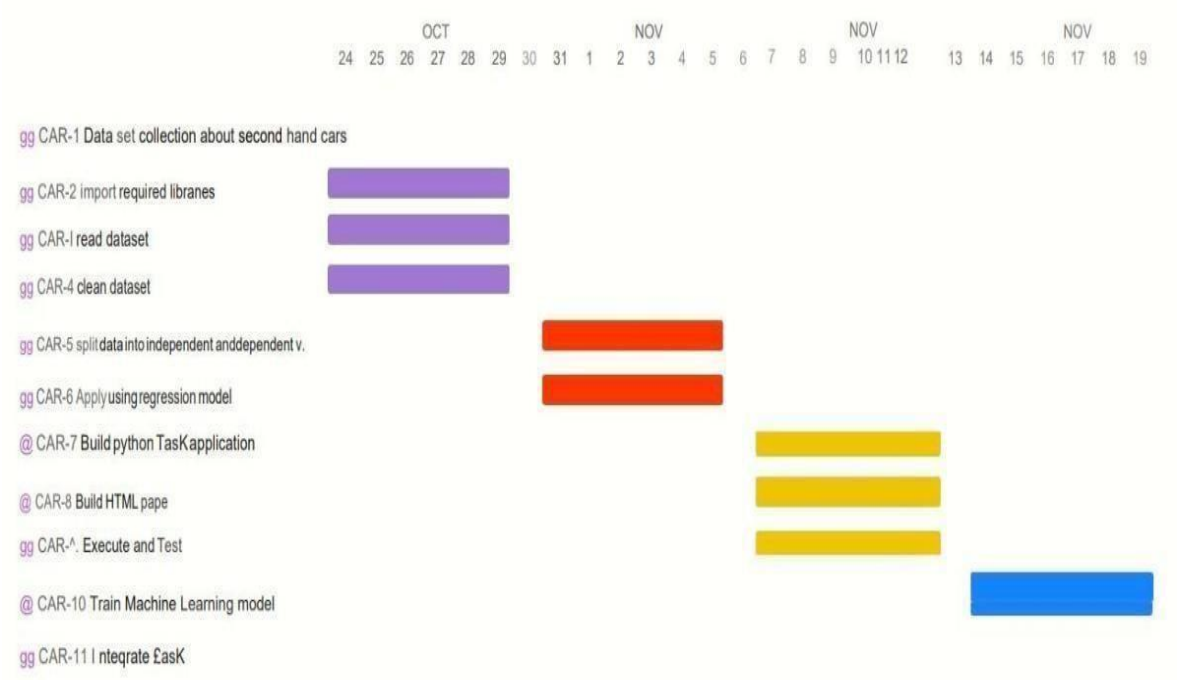
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per unit).let's calculate the team's averge velocity (AV) per iteration unit (story points per day).

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

6.3 Reports from JIRA

Burndown Chart:

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



Customer Journey Map:

7. CODING AND SOLUTIONING (Explain the features added in the project along with code)

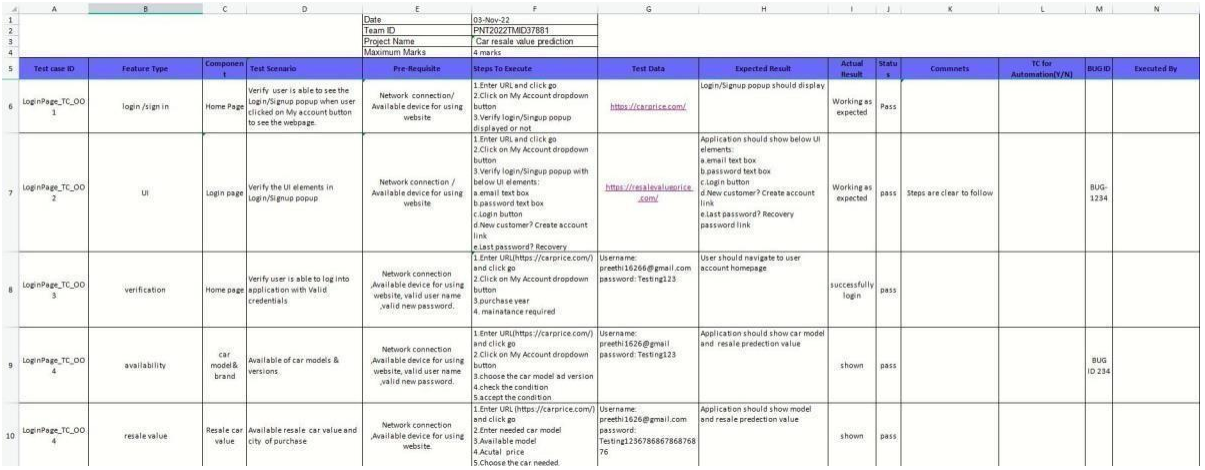
7.1 Feature 1

7.2 Feature 2

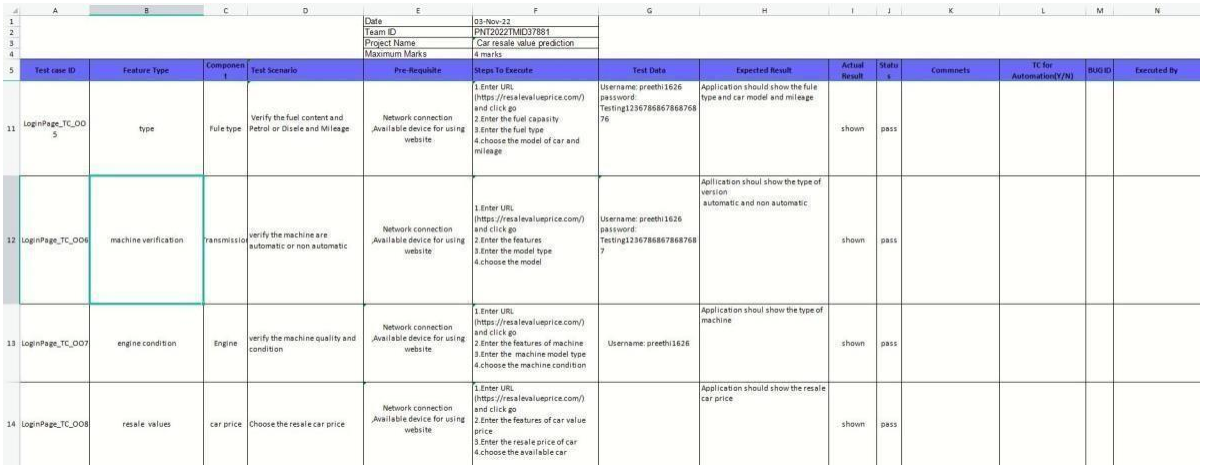
7.3 Database Schema (if Applicable)

8. TESTING

8.1 Test Cases



				Date	03-Nov-22								
				Team ID	PNT2022TMID37881								
				Project Name	Car resale value prediction								
				Maximum Marks	4 marks								
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO1	login /sign in	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button to see the webpage.	Network connection/ Available device for using website	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup displayed or not	https://carprice.com/	Login/Signup popup should display	Working as expected	Pass				
LoginPage_TC_OO2	UI	Login page	Verify the UI elements in Login/Signup popup	Network connection / Available device for using website	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Singup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery	https://resalevalueprice.com/	Application should show below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e.Last password? Recovery password link	Working as expected	pass	Steps are clear to follow		BUG-1234	
LoginPage_TC_OO3	verification	Home page	Verify user is able to log into application with Valid credentials	Network connection ,Available device for using website, valid user name ,valid new password.	1.Enter URL(https://carprice.com/) and click go 2.Click on My Account dropdown button 3.purchase year 4. mainatance required	Username: preethi1626@gmail.com password: Testing123	User should navigate to user account homepage	successfully login	pass				
LoginPage_TC_OO4	availability	car model& brand	Available of car models & versions.	Network connection ,Available device for using website, valid user name ,valid new password.	1.Enter URL(https://carprice.com/) and click go 2.Click on My Account dropdown button 3.choose the car model ad version 4.check the condition 5.accept the condition	Username: preethi1626@gmail password: Testing123	Application should show car model and resale predection value	shown	pass			BUG ID 234	
LoginPage_TC_OO4	resale value	Resale car value	Available resale car value and city of purchase	Network connection ,Available device for using website.	1.Enter URL (https://carprice.com/) and click go 2.Enter needed car model 3.Available model 4.Acutal price 5.Choose the car needed	Username: preethi1626@gmail.com password: Testing12367868678687676	Application should show model and resale predection value	shown	pass				



				Date	03-Nov-22								
				Team ID	PNT2022TMID37881								
				Project Name	Car resale value prediction								
				Maximum Marks	4 marks								
Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets	TC for Automation(Y/N)	BUG ID	Executed By
LoginPage_TC_OO5	type	Fule type	Verify the fuel content and Petrol or Disele and Mileage	Network connection ,Available device for using website	1.Enter URL (https://resalevalueprice.com/) and click go 2.Enter the fuel capasity 3.Enter the fuel type 4.choose the model of car and mileage	Username: preethi1626 password: Testing12367868678687676	Application should show the fule type and car model and mileage	shown	pass				
LoginPage_TC_OO6	machine verification	Transmission	verify the machine are automatic or non automatic	Network connection ,Available device for using website	1.Enter URL (https://resalevalueprice.com/) and click go 2.Enter the features 3.Enter the model type 4.choose the model	Username: preethi1626 password: Testing12367868678687677	Apllication shoul show the type of version automatic and non automatic	shown	pass				
LoginPage_TC_OO7	engine condition	Engine	verify the machine quality and condition	Network connection ,Available device for using website	1.Enter URL (https://resalevalueprice.com/) and click go 2.Enter the features of machine 3.Enter the machine model type 4.choose the machine condition	Username: preethi1626	Application shoul show the type of machine	shown	pass				
LoginPage_TC_OO8	resale values	car price	Choose the resale car price	Network connection ,Available device for using website	1.Enter URL (https://resalevalueprice.com/) and click go 2.Enter the features of car value price 3.Enter the resale price of car 4.choose the available car		Application should show the resale car price	shown	pass				

8.2 User Acceptance Testing

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

Section	Total Cases	Not Tested	Fail	Pass
Login /sign in	1	0	0	1
User interface	1	0	0	1
Availability	1	0	0	1
type	1	0	0	1
condition	1	0	0	1
verification	2	0	0	2
Resale price	2	0	0	2

9. RESULTS

9.1 Performance Metrics

Car Resales Price Prediction

MODEL BUILDING

Choose the metrics of the model

```
#predicting the values to test set
y_pred = regressor.predict(X_test)
#printing the accuracy for test set
print(r2_score(Y_test,y_pred))
```

10. ADVANTAGES AND DISADVANTAGES

Advantages

1. Application is easy to use
2. User Friendly
3. No Cost
4. No need to commission any agent to get car resale value estimate

Disadvantages

5. User needs to fill every asked detail of the car
6. Doesn’t work for cars from different distributions
7. Not always accurate

11. CONCLUSION

Car price prediction can be a challenging task due to the high number of attributes that should be considered for the accurate prediction. The major step in the prediction process is collection and preprocessing of the data. In this research, PHP scripts were built to normalize, standardize and clean data to avoid unnecessary noise for machine learning algorithms.

Data cleaning is one of the processes that increases prediction performance, yet insufficient for the cases of complex data sets as the one in this research.Applying single machine algorithm on the data set accuracy was less than 50%. Therefore, the ensemble of multiple machine learning algorithms has been proposed and this combination of ML methods gains accuracy of 92.38%. This is significant improvement compared to single machine learning method approach. However, the drawback of the proposed system is that it consumes much more computational resources than single machine learning algorithm. Although, this system has achieved astonishing performance in car price prediction problem our aim for the future research is to test this system to work successfully with various data sets. We will extend our test data with eBay [16] and OLX [17] used cars data sets and validate the proposed approach.

12. FUTURE SCOPE

For better performance, we plan to judiciously design deep learning network structures, use adaptive learning rates and train on clusters of data rather than the whole dataset. To correct for overfitting in Random Forest, different selections of features and number of trees will be tested to check for change in performance.

13. APPENDIX

Source Code

Car Resale Value Prediction.ipynb

#Import libraries

```
import datetime

import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import r2_score
```

#Read dataset

```
dataset = pd.read_csv("data/dataset.csv")
dataset.head(5)

X_train, X_test, y_train, y_test = train_test_split(dataset.iloc[:, :-1],
                                                    dataset.iloc[:, -1],
                                                    test_size = 0.3,
                                                    random_state = 42)
```

```
X_train.info()
```

#Index

```
X_train = X_train.iloc[:, 1:]
X_test = X_test.iloc[:, 1:]
```

#Name

```
X_train["Name"].value_counts()

make_train = X_train["Name"].str.split(" ", expand = True)
make_test = X_test["Name"].str.split(" ", expand = True)

X_train["Manufacturer"] = make_train[0]
X_test["Manufacturer"] = make_test[0]

plt.figure(figsize = (12, 8))
plot = sns.countplot(x = 'Manufacturer', data = X_train)
plt.xticks(rotation = 90)
for p in plot.patches:
    plot.annotate(p.get_height(),
                  (p.get_x() + p.get_width() / 2.0,
                   p.get_height()),
                  ha = 'center',
                  va = 'center',
                  xytext = (0, 5),
                  textcoords = 'offset points')

plt.title("Count of cars based on manufacturers")
plt.xlabel("Manufacturer")
plt.ylabel("Count of cars")

X_train.drop("Name", axis = 1, inplace = True)
X_test.drop("Name", axis = 1, inplace = True)
```

#Location

```
X_train.drop("Location", axis = 1, inplace = True)
X_test.drop("Location", axis = 1, inplace = True)
```

#Year

```
curr_time = datetime.datetime.now()
X_train['Year'] = X_train['Year'].apply(lambda x : curr_time.year - x)
```

```
X_test['Year'] = X_test['Year'].apply(lambda x : curr_time.year - x)

#Kilometers_Driven

X_train["Kilometers_Driven"]
```

#Mileage

```
mileage_train = X_train["Mileage"].str.split(" ", expand = True)
mileage_test = X_test["Mileage"].str.split(" ", expand = True)

X_train["Mileage"] = pd.to_numeric(mileage_train[0], errors = 'coerce')
X_test["Mileage"] = pd.to_numeric(mileage_test[0], errors = 'coerce')

print(sum(X_train["Mileage"].isnull()))
print(sum(X_test["Mileage"].isnull()))

X_train["Mileage"].fillna(X_train["Mileage"].astype("float64").mean(), inplace =
True)
X_test["Mileage"].fillna(X_train["Mileage"].astype("float64").mean(), inplace =
True)
```

#Engine, Power and Seats

```
cc_train = X_train["Engine"].str.split(" ", expand = True)
cc_test = X_test["Engine"].str.split(" ", expand = True)
X_train["Engine"] = pd.to_numeric(cc_train[0], errors = 'coerce')
X_test["Engine"] = pd.to_numeric(cc_test[0], errors = 'coerce')

bhp_train = X_train["Power"].str.split(" ", expand = True)
bhp_test = X_test["Power"].str.split(" ", expand = True)
X_train["Power"] = pd.to_numeric(bhp_train[0], errors = 'coerce')
X_test["Power"] = pd.to_numeric(bhp_test[0], errors = 'coerce')

X_train["Engine"].fillna(X_train["Engine"].astype("float64").mean(), inplace =
True)
X_test["Engine"].fillna(X_train["Engine"].astype("float64").mean(), inplace =
True)

X_train["Power"].fillna(X_train["Power"].astype("float64").mean(), inplace =
True)
X_test["Power"].fillna(X_train["Power"].astype("float64").mean(), inplace = True)

X_train["Seats"].fillna(X_train["Seats"].astype("float64").mean(), inplace = True)
X_test["Seats"].fillna(X_train["Seats"].astype("float64").mean(), inplace = True)
```

#New Price

```
X_train.drop(["New_Price"], axis = 1, inplace = True)
X_test.drop(["New_Price"], axis = 1, inplace = True)

#Data Processing

X_train = pd.get_dummies(X_train,
                         columns = ["Manufacturer", "Fuel_Type", "Transmission",
"Owner_Type"],
                         drop_first = True)

X_test = pd.get_dummies(X_test,
                         columns = ["Manufacturer", "Fuel_Type", "Transmission",
"Owner_Type"],
                         drop_first = True)

missing_cols = set(X_train.columns) - set(X_test.columns)
for col in missing_cols:
    X_test[col] = 0
X_test = X_test[X_train.columns]

standardScaler = StandardScaler()
standardScaler.fit(X_train)
X_train = standardScaler.transform(X_train)
X_test = standardScaler.transform(X_test)
```

#Training and predicting

```
linearRegression = LinearRegression()
linearRegression.fit(X_train, y_train)
y_pred = linearRegression.predict(X_test)
r2_score(y_test, y_pred)

rf = RandomForestRegressor(n_estimators = 100)
rf.fit(X_train, y_train)
y_pred = rf.predict(X_test)
r2_score(y_test, y_pred)
```

app.py

```
#app.py
from flask import Flask,request, url_for, redirect, render_template, jsonify,session
import sqlite3 as sql
from flask_cors import CORS, cross_origin
import pickle
import numpy as np
import os
import pandas as pd
import joblib

app = Flask(__name__)
app.secret_key = "Secret Key"
# load the saved model file and use for prediction
model = pickle.load(open("CarPricePredictionModel.pkl", "rb"))




@app.after_request # blueprint can also be app~~
def after_request(response):
    header = response.headers
    header['Access-Control-Allow-Origin'] = '*'
    return response


# ==================================
#  Insert data in database (SIGNUP)
# ==================================
def insertUser(username, email, password, contact):
    con = sql.connect("SignUP.db")
    cur = con.cursor()
    phone = int(contact)
    query = ("""INSERT INTO SignUP
            (username,email,password,contact)
            VALUES ('%s','%s','%s',%s)""" %
            (username, email, password, contact))
    cur.execute(query)
    con.commit()
    con.close()


# ====================================
```

```
#  Validating data in database (LOGIN)
# ====================================
def validUser(email, password):
    con = sql.connect("SignUP.db")
    cur = con.cursor()
    query = ("""SELECT * FROM SignUP
            where email = '%s' and password = '%s'
            """ %
            (email, password))
    cur.execute(query)
    data = cur.fetchall()
    con.close()
    return data


# ==================
#    Flask Routing
# ==================

@app.route('/')
def home111():
    return render_template('login_1.html')

# Login page
@app.route('/login_1', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        rd = validUser(request.form['email'], request.form['password'])
        if rd:
            session['user']=rd[0]
            return render_template('homepage_1.html')
        else:
            msg="Wrong username or password"
            return render_template('login_1.html',msg=msg)
    else:
        return render_template('login_1.html')


@app.route('/logout')
def logout1():
            session.pop('user', None)
            return render_template('login_1.html')


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

```
def student():
    if 'user' in session:
        s = session['user']
        all_data = Student.query.all()
        return render_template("homepage_1.html", all_data = all_data,user=s)
    else:
        return render_template('login_1.html')



# Signup page
@app.route('/signup/', methods=['GET', 'POST'])
def signup():
    if request.method == 'POST':
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        contact = request.form['contact']
        insertUser(username, email, password, contact)
        msg= "account created successfully"
        return redirect(url_for('login'))
    else:
        return render_template('login_1.html')

# api json
@app.route('/sum', methods=['GET','POST'])
def sum():
    sum = 0
    a = int(request.args.get('a'))
    b = int(request.args.get('b'))
    sum = a+b
    return jsonify(sum)


@app.route('/mainpage')
def mainhome():
    return render_template("homepage_1.html")


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

@app.route('/about')
def about():
```

```
    return render_template("about.html")


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

    if request.method == 'POST':
        year = int(request.form['year'])
        km_driven=float(request.form['km_driven'])
        owner=request.form['owner']
        if(owner=='test'):
            owner=0
        elif(owner=='first'):
            owner=1
        elif(owner=='second'):
            owner=2
        elif(owner=='third'):
            owner=3
        elif(owner=='fourth'):
            owner=4

        fuel=request.form['fuel']
        if(fuel=='Diesel'):
            fuel=0
        elif(fuel=='Petrol'):
            fuel=1
        elif(fuel=='LPG'):
            fuel=2
        elif(fuel=='CNG'):
            fuel=3
        Current_year = 2021
        years_driven = Current_year - year
        seller_type=request.form['seller_type']
        if(seller_type=='Individual'):
            seller_type=0
        elif(seller_type=='Dealer'):
            seller_type=1
        transmission=request.form['transmission']
        if(transmission == 'Mannual'):
            transmission=1
        elif(transmission == 'Automatic'):
            transmission=0
        mileage = float(request.form['mileage'])
        engine = float(request.form['engine'])
        max_power = float(request.form['max_power'])
```

```
        max_power = max_power - 30
        torque = float(request.form['torque'])
        torque = torque - 40
        seats = int(request.form['seats'])
        prediction=model.predict(np.array([[year, km_driven, fuel, seller_type,
transmission, owner, mileage, engine, max_power, torque, seats, Current_year,
years_driven]]))
        #output=round(prediction[0],2)
        output1  = str(prediction)
        output = output1.strip("[].")
        #if output<0:
        #    return render_template('index.html',prediction_texts="Sorry you cannot
sell this car")
        #else:
        return render_template('predict.html',prediction_text="You can sell the Car at
₹{}".format(output))
    else:
        return render_template('predict.html')



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

HTML:

1. **Login.html**

```
<html>
<head>
        <title>Login and Registration form</title>
         <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="{{url_for("static",filename="css/style.css")}}">

</head>
<body>
<div class="container">
<div class="card">
<div class="inner-box" id="card">
<div class="card-front">
        <h2>LOGIN</h2>
        <form action="/login_1" method="post">
```

```
      <input type="email" name="email" class="input-box" placeholder="Your
    Email Id" required><br><br>
      <input type="password" class="input-box" name="password"
    placeholder="Password" required><br><br>
      {{ msg }}
      <button type="submit" value="Login" class="submit">Lets Drive</button>
    </form>
    <br>
    <br>
    <h4 align="center">Haven't Registered Yet..! </h4>
    <button type="button" class="btn" onclick="openRegister()">Click here to
    Register</button>

</div>



<div class="card-back">
    <h2>REGISTER</h2>
    <form action="/signup" method="post">
      <input type="text" class="input-box" name="username"
    placeholder="Your name" required>
      <input type="email" class="input-box" name="email" placeholder="Your
    Email Id" required>
      <input type="password" class="input-box" name="password"
    placeholder="Password" required>
      <input type="tel" name="contact" class="input-box" placeholder="contact-
    no" required>
      <button type="submit" class="submit"> Submit</button>
    </form>
    <button type="button" class="btn" onclick="openlogin()">I have an account
    > LOGIN</button>
</div>
</div>
</div>
</div>
</body>
<script defer>

    var card = document.getElementById("card")


    function openRegister(){
      card.style.transform= "rotateY(-180deg)";
    }
```

```
    function openlogin(){
      card.style.transform= "rotateY(0deg)";
    }
</script>
    </html>
```

2. **Homepage.html**

```
    <!DOCTYPE html>
    <html>

    <head>
      <title>
        Car Price Prediction
      </title>
        <meta name="viewport" content="width=device-width, initial-scale=1">
    <style>
    body {
        background-size: cover;
        background-repeat: no-repeat;
        background-attachment:fixed;
        background-position: center;
    }
      margin: 0;
      font-family: Arial;
    }

    .topnav
    {
      position: absolute;
      overflow: hidden;
      background-color: blue;
    }

    .topnav a {
      float: top-center;
      position: relative;
      color: black;
      text-align: center;
      padding: 14px 16px;
      text-decoration: none;
      font-size: 40px;

    }
```

```
.topnav a:hover {
  background-color: lightgrey;
  color: black;
}

.topnav a.active {
   color: black;
}


</style>
<title>Iqbal's Website</title>
</head>

<body div style="background-image: url('/static/car7.jpg');">


<div class="topnav">
  <div class="row">

  <div style="display: inline-block;padding: 30px"><a class="active"
href="#">Home</a></div>
  <div style="display: inline-block;padding: 30px"><a
href="/predict">Predict</a></div>
  <div style="display: inline-block;padding: 30px"><a
href="/logout">Logout</a></div>


</div>

<div style="color: black; font-size: 40px; text-align: center;"><h2>Car Price
Prediction</h2></div>
<div style="color: white; font-size: 50px; text-align: center;"><h5> Eager to
know the Re-Sale Value of your Car...Then You are at the Right
Place</h5></div>
<br>
<br>
<br>
<br>
<br>
<br>
<br>
<marquee>
<font color=white size="2">
```

```
   <h1> This is a Project for Estimating the Resale Value of any Car and
works best for Cars ranging from ₹ 50,000/- to ₹50,00,000/-  </h1>
    </font>
</marquee>
<br>
<br>
</body>

</html>
```

3. Predict.html

```
<html>

<head>
<style>
body {
    background-size: cover;
    background-repeat: no-repeat;
    background-attachment:fixed;
    background-position: center;
}
  margin: 0;
  font-family: Arial;
}

.topnav
{
  overflow: hidden;
  background-color: blue;
}

.topnav a {
  float: top-center;

  color: black;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
  font-size: 40px;

}

.topnav a:hover {
  background-color: lightgrey;
```

```
  color: black;
}

.topnav a.active {
   color: black;
}


</style>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Car Price Prediction</title>


    <!-- BootStrap -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.c
ss"
        integrity="sha384-
9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcP
b1dKGj7Sk" crossorigin="anonymous">


    <!-- css -->
    <link rel="stylesheet" href="static/css/styles.css">


</head>

<body div style="background-image: url('static/car.jpg');">

<div class="topnav">
  <div class="row">

  <div style="display: inline-block;padding: 30px"><a class="active"
href="/mainpage">Home</a></div>
  <div style="display: inline-block;padding: 30px"><a
href="/predict">Predict</a></div>
  <div style="display: inline-block;padding: 30px"><a
href="/logout">Logout</a></div>

</div>

    <!-- As a heading -->
```

```
        <div class="navbar-header">

            <h3 align="center" href="/">Car Re-Sale Price Prediction </h3>

        </div>


    <div class="container-fluid">


        <p style="font-size: 12px; text-align: center;">(Please fill the parameters below and click on Selling Price button. Scroll to the last to check car price after clicking on Selling Price button)</p>
        <p style="font-size: 12px; text-align: center;">*NOTE* - Enter Numeric Values only</p>

        <div class="row">
        <form action="\predict" method="post">

            <div style="display: inline-grid;padding: 45px "><h3>Year of Buying</h3><input id="first" name="year" placeholder="eg. like the year '2010' "type="number "></div>

            <div style="display: inline-grid;padding: 45px "><h3>Kilometers Driven</h3><input id="third" name="km_driven" placeholder="eg. 10000 Km driven before"required="required"></div>


            <div style="display: inline-grid;padding: 45px "><h3>Ownership</h3><select name="owner" id="fourth" required="required">
                <option value="first">First Owner</option>
                <option value="second">Second Owner</option>
                <option value="third">Third Owner</option>
                <option value="fourth">Fourth Owner</option>
            </select></div>


            <div style="display: inline-grid;padding: 45px "><h3>Fuel Type</h3><select name="fuel" id="fuel" required="required">
                <option value="Diesel">Diesel</option>
                <option value="Petrol">Petrol</option>
        <option value="CNG">CNG</option>
            </select></div>
```

```
        <div style="display: inline-grid;padding: 45px "><h3>Dealer or
Individual</h3><select name="seller_type" id="resea" required="required">
            <option value="Individual">Individual</option>
        <option value="Dealer">Dealer</option>
      </select></div>

        <div style="display: inline-grid;padding: 45px "><h3>Transmission
Type</h3><select name="transmission" id="research" required="required">
            <option value="Mannual">Manual Car</option>
            <option value="Automatic">Automatic Car</option>
        </select></div>


   <div style="display: inline-grid;padding: 45px "><h3>Mileage
(kmpl)</h3><input id="first" name="mileage" placeholder="btw 5 to
50"type="number " ></div>

   <div style="display: inline-grid;padding: 45px "><h3>Engine
(cc)</h3><input id="first" name="engine" placeholder="600 -
3600"type="number " ></div>

   <div style="display: inline-grid;padding: 45px "><h3>Max Power
(bhp)</h3><input id="first" name="max_power" placeholder="30 -
300"type="number " ></div>

   <div style="display: inline-grid;padding: 45px "><h3>Torque
(Nm)</h3><input id="first" name="torque" placeholder="50 -
700"type="number " ></div>

   <div style="display: inline-grid;padding: 45px "><h3>Seats</h3><input
id="first" name="seats" placeholder="2-9"type="number " ></div>
   <br>

   <div style="float: none; text-align: center;padding: 1px "><button
class="submit" type="submit ">Click Here to Find The Selling
Price</button></div></div>
      </form>
      <div style="display: inline-grid;padding: 45px "><h3>{{
prediction_text }}</h3></div>
      <br>
   </div>
   </div>
```

```
    <!-- JavaScript -->
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"
        integrity="sha384-
DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+Or
CXaRkfj"
        crossorigin="anonymous"></script>
    <script
src="https://cdn.jsdelivr.net/npm/popper.js@1.16.0/dist/umd/popper.min.js"
        integrity="sha384-
Q6E9RHvbIyZFJoft+2mJbHaEWldlvI9IOYy5n3zV9zzTtmI3UksdQRVvox
MfooAo"
        crossorigin="anonymous"></script>
    <script
src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/js/bootstrap.min.js"
        integrity="sha384-
OgVRvuATP1z7JjHLkuOU7Xw704+h835Lr+6QL9UvYjZE3Ipu6Tp75j7Bh
/kR0JKI"
        crossorigin="anonymous"></script>
    <script src="https://kit.fontawesome.com/5f3f547070.js"
crossorigin="anonymous"></script>

   <!-- Footer -->


</body>

</html>
```

CSS:

```
*{
        margin:0;
        padding:0;
}
.container{
        width: 100%;
        height: 100vh;
        font-family: sans-serif;
        background-image: url("/static/car13.jpg");
        width=100%;
        display: flex;
        align-items: center;
```

```
        justify-content: center;
}
.card{
        width: 350px;
        height: 500px;
    box-shadow: 0 0 40px 20px rgba(0,0,0,0.26);
        perspective: 1000px;
}
.inner-box{
        position: relative;
        width: 100%;
        height: 100%;
        transform-style: preserve-3d;
        transition: transform 1s;

}
.card-front, .card-back{
        position: absolute;
        width: 100%;
        height: 100%;
        background-position: center;
        background-size: cover;
        background-image: linear-gradient(rgba(84, 201, 86, 0.7),rgba(84, 201,
86, 0.7)),url(background.png);
        padding: 55px;
        box-sizing: border-box;
        backface-visibility: hidden;
}
.card-back{
        transform: rotateY(180deg);
}
.card h2{
        font-weight: normal;
        font-size: 24px;
        text-align: center;
        margin-bottom: 20px;
}
.input-box{
        width: 100%;
        background: transparent;
        border: 1px solid #fff;
        margin: 6px 0;
        height: 32px;
        border-radius: 20px;
        padding: 0 10px;
```

```
        box-sizing: border-box;
        outline: none;
        text-align: center;
        color: #fff;
}
::placeholder{
        color: black;
        font-size: 12px;
}
button{
        width: 100%;
        background: transparent;
        border: 1px solid #fff;
        margin: 35px 0 10px;
        height: 32px;
        font-size: 12px;
        border-radius: 20px;
        padding: 0 10px;
        box-sizing: border-box;
        outline: none;
        color: #fff;
        cursor: pointer;
}
.submit-btn{
        position: relative;
}
.submit-btn::after{
        content: '\27a4';
        color: #333;
        line-height: 32px;
        font-size:17px;
        height: 32px;
        width: 32px;
        border-radius: 50%;
        background: #fff;
        position: absolute;
        right: -1px;
        top: -1px;
}
span{
        font-size: 13px;
        margin-left: 10px;
}
```

```
.card .btn{

    margin-top: 70 px;
}
.card a{
        color: #fff;
        text-decoration: none;
        display: block;
        text-align: center;
        font-size: 13px;
        margin-top: 8px;
}
```

GitHub Link:

https://github.com/IBM-EPBL/IBM-Project-11196-1659276167

Project Demo Link