PROJECT-BASED EXPERIENTIAL LEARNING PROGRAM

(NALAIYA THIRAN)

PROJECT REPORT

CAR RESALE VALUE PREDICTION

PNT2022TMID13785

TEAM MEMBERS:

711319CS001 - ABHISHEK V S
711319CS007 - AKASH G
711319CS015 - ARAVIND KARTI S
711319CS050 - HARSHINI R

Index

6. **PROJECT PLANNING & SCHEDULING**
    1. Sprint Planning & Estimation
    2. Sprint Delivery Schedule
    3. Reports from JIRA

7. **CODING & SOLUTIONING (Explain the features added in the project along with code)**
    1. Feature 1
    2. Feature 2
    3. Database Schema (if Applicable)

8. **TESTING**
    1. Test Cases
    2. User Acceptance Testing

9. **RESULTS**
    1. Performance Metrics

10. **ADVANTAGES & DISADVANTAGES**

11. **CONCLUSION**

12. **FUTURE SCOPE**

13. **APPENDIX**

    Source Code

    GitHub & Project Demo Link

Car Resale Value Prediction

1. INTRODUCTION

It is expected that sales of old automobiles and second-hand imported (refurbished) autos would rise in tough economic times. Leasing an automobile rather than purchasing one entirely is typical in many affluent nations. After the lease term is up, the buyer will have the option of purchasing the vehicle for its residual value, or anticipated resale value. Therefore, being able to accurately anticipate the salvage value (residual value) of automobiles is in the best interest of sellers and financiers from a business standpoint.

1. Project Overview

We proposed an intelligent, adaptable, and successful approach based on applying regression algorithms to forecast the car's resale value. An estimation of the vehicle's resale value must be made using a regression model that takes into account the key variables that influence its value. The method with the highest accuracy will be chosen from among the several regression algorithms we employ, and after that it will be implemented into the web-based application that notifies the user of the status of his product.

2. Purpose

The reliability of the automobile may be successfully assessed using a range of criteria, which is why a used car price prediction system is required. The prediction approach used by certain websites, despite the fact that they offer this service, may not be the finest. Additionally, many methods and algorithms could help with the accuracy of predictions for the real market value of a used automobile. In order to make informed decisions while purchasing or selling, one must be aware of their true market value.

2. LITERATURE SURVEY

1. Existing problem

With certain additional costs imposed by the Government in the form of taxes, the manufacturer sets the prices of new cars in the market. Customers who purchase a new car can be sure that their investment will be worthwhile. Used car sales, however, are rising globally as a result of the rising cost of new cars and the inability of consumers to purchase new cars due to a lack of cash.
To accurately assess the value of the car using a number of features, a used car price prediction system is required. Although there are websites that offer this service, their prediction method might not be the best. In addition, various models and systems may help predict the true market value of a used car. When buying or selling, it's crucial to understand their true market value.

2. References

[1] Kanwal Noor, 2017, Vehicle Price Prediction System using Machine Learning Techniques International Journal of Computer Applications. Volume 167 - Number 9

[2] Mariana Lusitania et al, (2009). Support vector regression analysis for price prediction in a vehicle leasing application

[3] Richardson, M. S. (2009). Determinants of used vehicle resale value.

[4] Listiani, M. (2009). Support vector regression analysis for price prediction in a car leasing application (Doctoral dissertation, Master thesis, TU Hamburg-Harburg).

[5] Richardson, M. S. (2009). Determinants of used car resale value. Retrieved from: https://digitalcc.coloradocollege.edu/islandora/object /

[6] Wu, J. D., Hsu, C. C., & Chen, H. C. (2009). An expert system of price forecasting for used cars using adaptive neuro-fuzzy inference. Expert Systems with Applications, 36(4), 7809-7817. [7] Gongqi, S., Yansong, W., &Qiang, Z. (2011, January). New Model for Residual Value Prediction of the Used Car Based on BP Neural Network and Nonlinear Curve Fit. In Measuring Technology and Mechatronics Automation (ICMTMA), 2011 Third International Conference on (Vol. 2, pp. 682).

[8]. Pudaruth, S. (2014). Predicting the price of used cars using machine learning techniques. Int. J.Inf.Comput. Technol, 4(7), 753-764.

[9]. Noor, K., & Jan, S. (2017). Vehicle Price Prediction System using Machine Learning Techniques. International Journal of Computer Applications, 167(9), 27-31.

[10] Weka 3 - Data Mining with Open Source Machine Learning Software in Java. (n.d.)

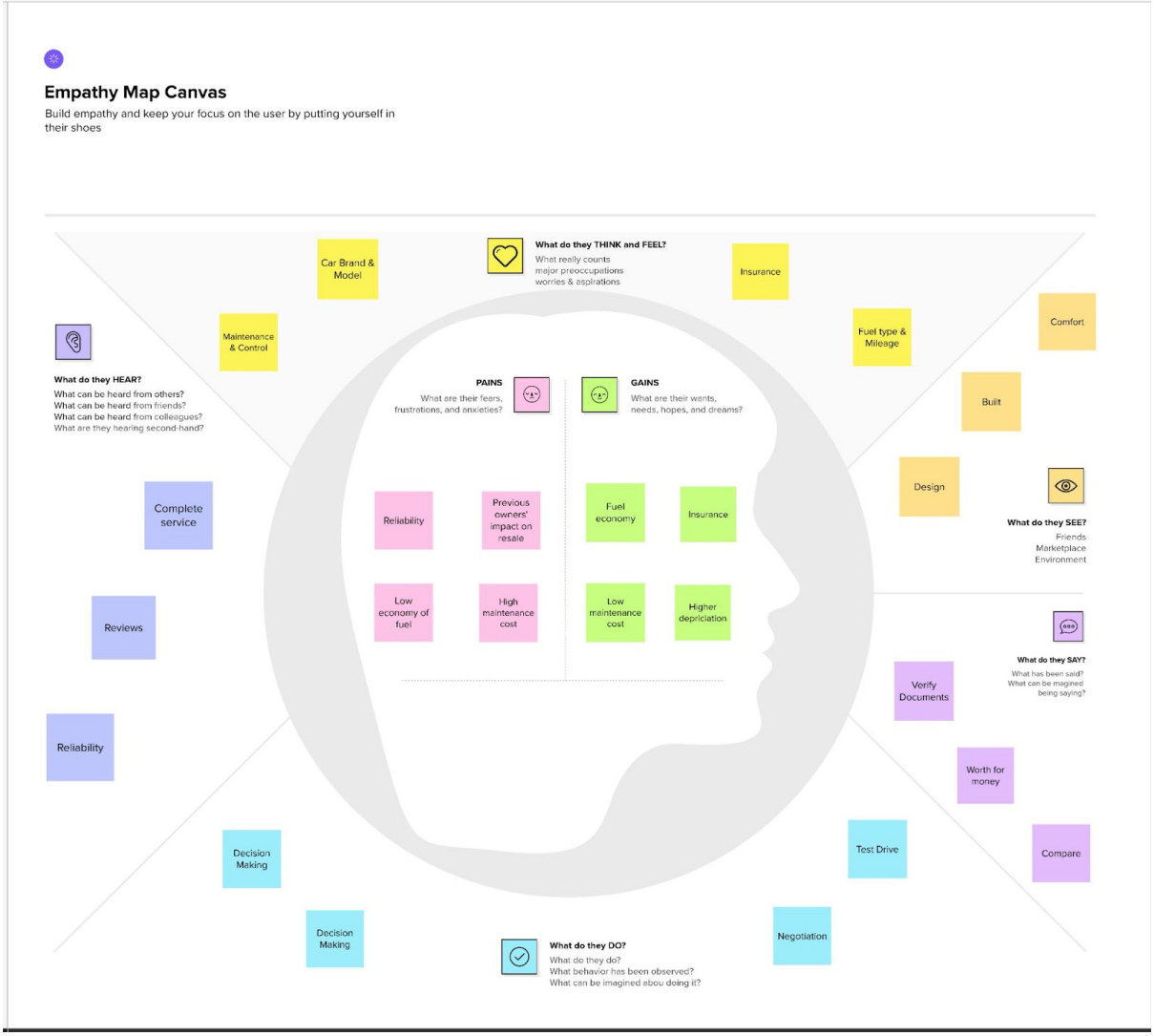
3. Problem Statement Definition

Almost all tasks that involve estimation now employ machine learning as a tool. Regression analysis is used by businesses like Cars24 and Cardekho.com to estimate used car prices. Therefore, we must create a model to calculate the cost of used cars. The model should input parameters related to cars and produce a selling price. The following characteristics are what determine a used car's selling price:

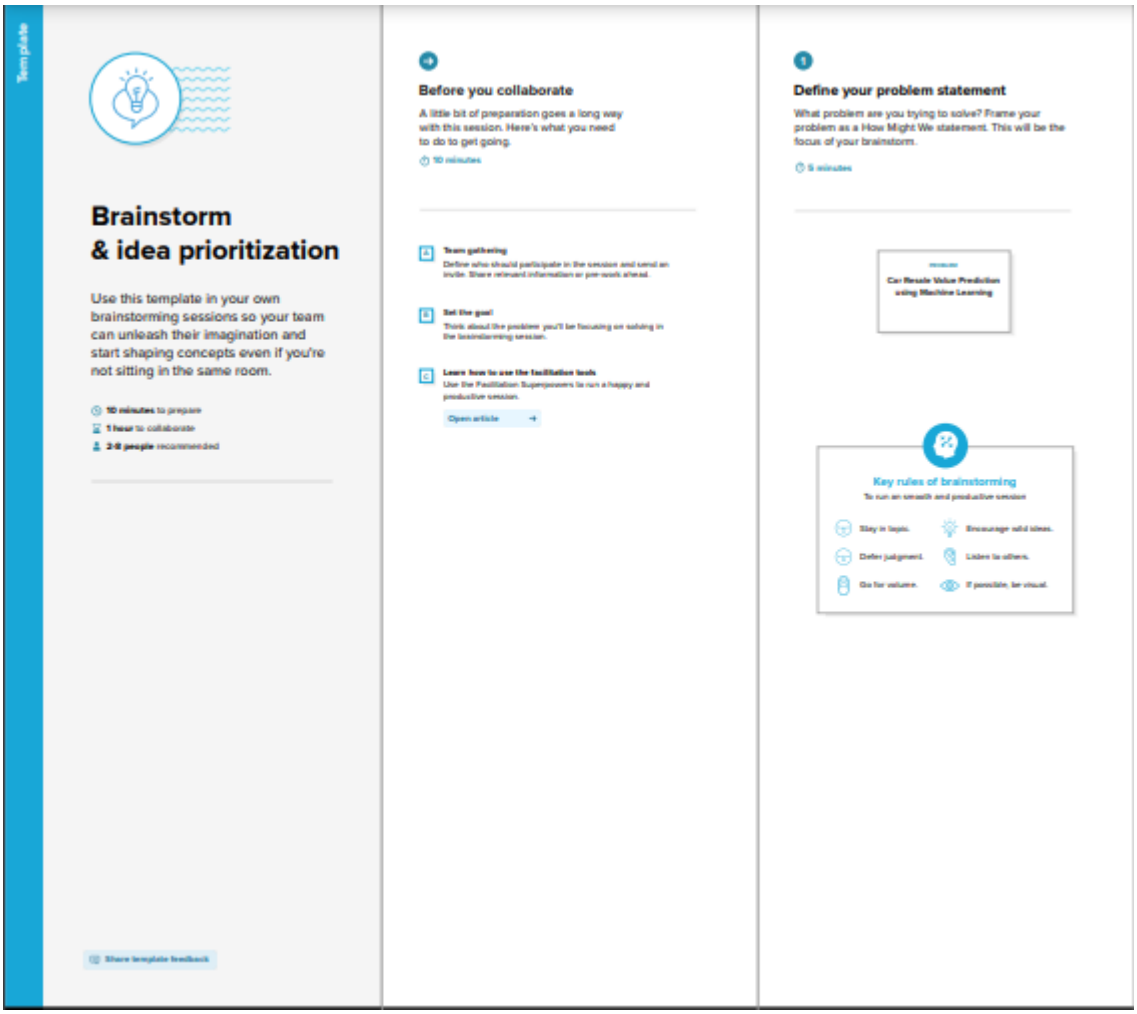
- Fuel Type
- Manufacturing Year
- Miles Driven
- Number of Historical Owners
- Maintenance Record

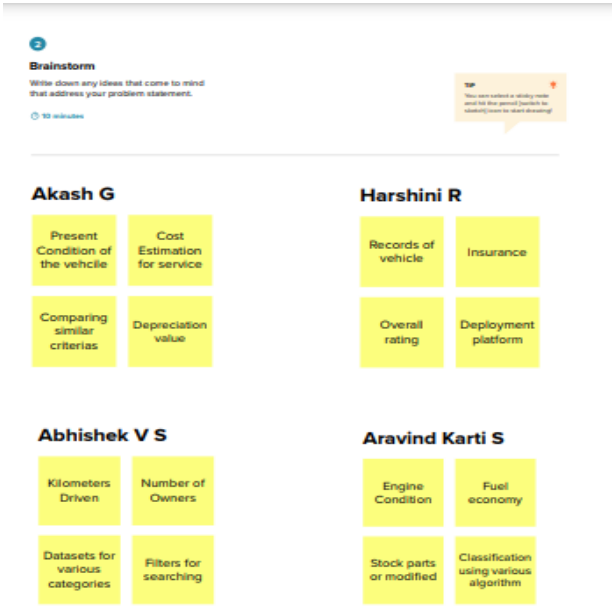
3. IDEATION & PROPOSED SOLUTION

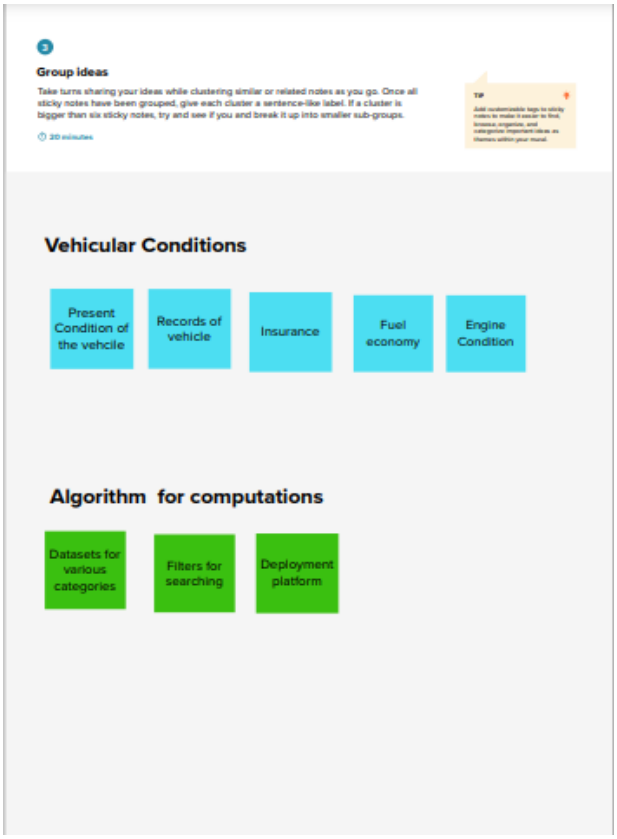
1. Empathy Map Canvas



2. Ideation & Brainstorming




2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes
TIP
Akash G
Present Condition of the vehcile
Cost Estimation for service
Comparing similar criterias
Depreciation value
Harshini R
Records of vehicle
Insurance
Overall rating
Deployment platform
Abhishek V S
Kilometers Driven
Number of Owners
Datasets for various categories
Filters for searching
Aravind Karti S
Engine Condition
Fuel economy
Stock parts or modified
Classification using various algorithm


3
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
TIP
Vehicular Conditions
Present Condition of the vehcile
Records of vehicle
Insurance
Fuel economy
Engine Condition
Algorithm for computations
Datasets for various categories
Filters for searching
Deployment platform

4

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.

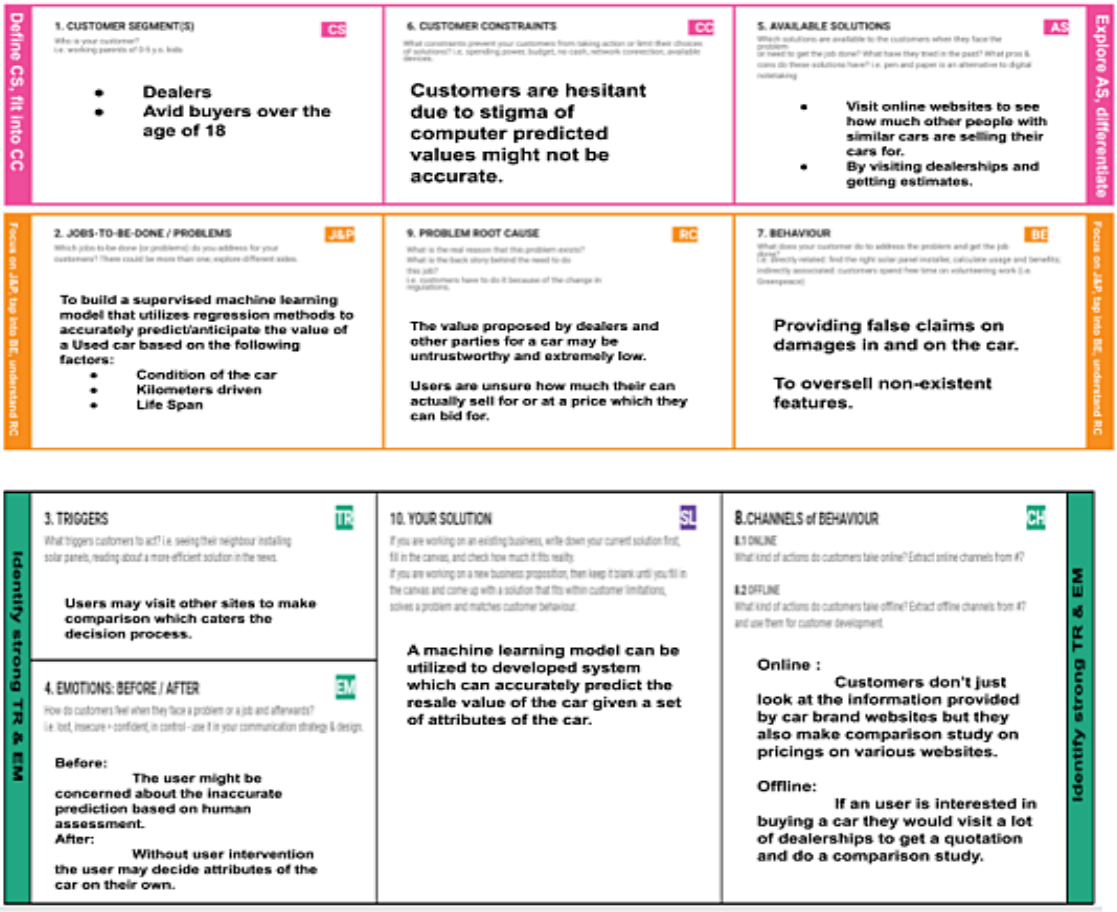
20 minutes

3. Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	• Sales forecasting is a current numerous trend in which all business companies thrive, and it also assists the organization or concern in determining its future goals and its plan and procedure to achieve them. • Before purchasing it, various factors such as, engine condition, company service record, spare parts condition, kilometres covered, mileage, number of owners, and engine condition are considered.
2.	Idea / Solution description	1. The overall concept proposed is to predict the resale value of a car and show it to the appropriate people. 2. This concept can be implemented and presented to the customer. There are two stages to this. 3. The first phase involves gathering data for training the car resale value prediction model. 4. Putting the car resale value prediction model to the test. 5. The second phase entails developing a website (front end) for presenting the entire solution as a customized GUI, making it very useful for the user to use this solution. The user will be asked to enter details for prediction such as model, price, built, and colour. If the user clicks the predict option, the predicted resale value will be displayed in the website
3.	Novelty / Uniqueness	• Consumer behaviour does in fact alter. Choose a product that has been added more recently wherever possible for improved accuracy. • The novelty sales predictions will be based on features from all of them using the average, and you can utilise many reference goods to acquire the best average.
4.	Social Impact / Customer Satisfaction	• If the user wants to buy or sell an 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 assists users in predicting the accurate value of an automobile remotely with perfect valuation and without the involvement of humans, such as car dealers, in the process to remove skewed value predictions made by dealers.
6.	Scalability of the Solution	This study offered a scalable system for estimating values for various types of old automobiles present throughout India using stored data and machine learning methodologies.

4. **Problem Solution fit**

Project Title:CAR RESALE VALUE PREDICTION
Team ID:PNT2022TMID13785



4. REQUIREMENT ANALYSIS

1. Functional requirement

Functional Requirements:

The suggested solution's functional requirements are listed below.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User entering Details	Enter Registration Number Enter the Specifications Details
FR-2	Data Visualization and Data Pre-processing	Performs visualization via matplotlib Performs visualization via seaborn Performs pre-processing via NumPy Performs pre-processing via pandas
FR-3	Implementing Machine Learning Algorithms	Implementing Regression algorithms
FR-4	Evaluate Prediction	Analyse the dataset's specifics using the built-in model.

2. Non-Functional requirements

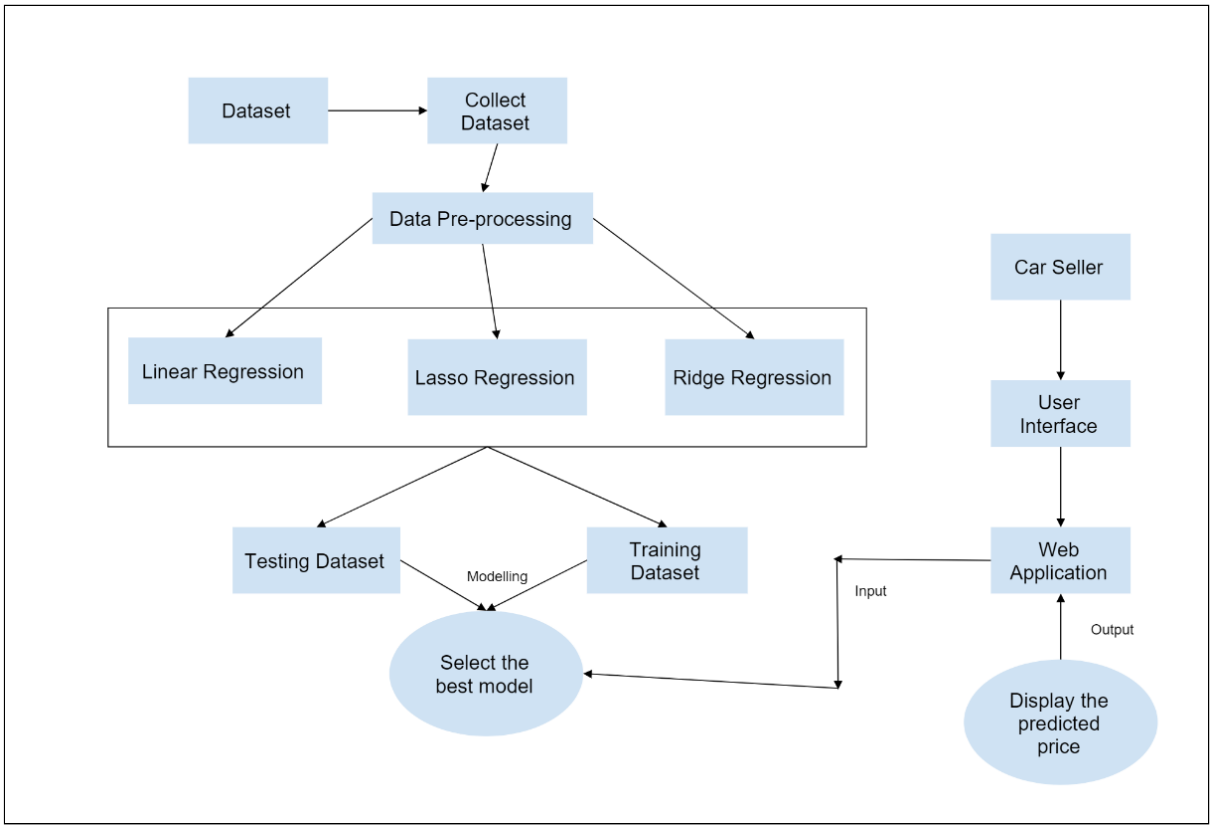
Non-functional Requirements:

The non-functional prerequisites of the suggested remedy are listed below.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The application is helpful in estimating the used car's worth with a high degree of accuracy.
NFR-2	**Security**	Only the user and the admin can view details to secure the details
NFR-3	**Reliability**	The forecast is accurate because of the better precision of the ML model.
NFR-4	**Performance**	Reduce the overall load time of prediction
NFR-5	**Availability**	The website is accessible to everyone, everywhere, and at any time.
NFR-6	**Scalability**	Managing workload

5. PROJECT DESIGN

1. Data Flow Diagrams



2. Solution & Technical Architecture

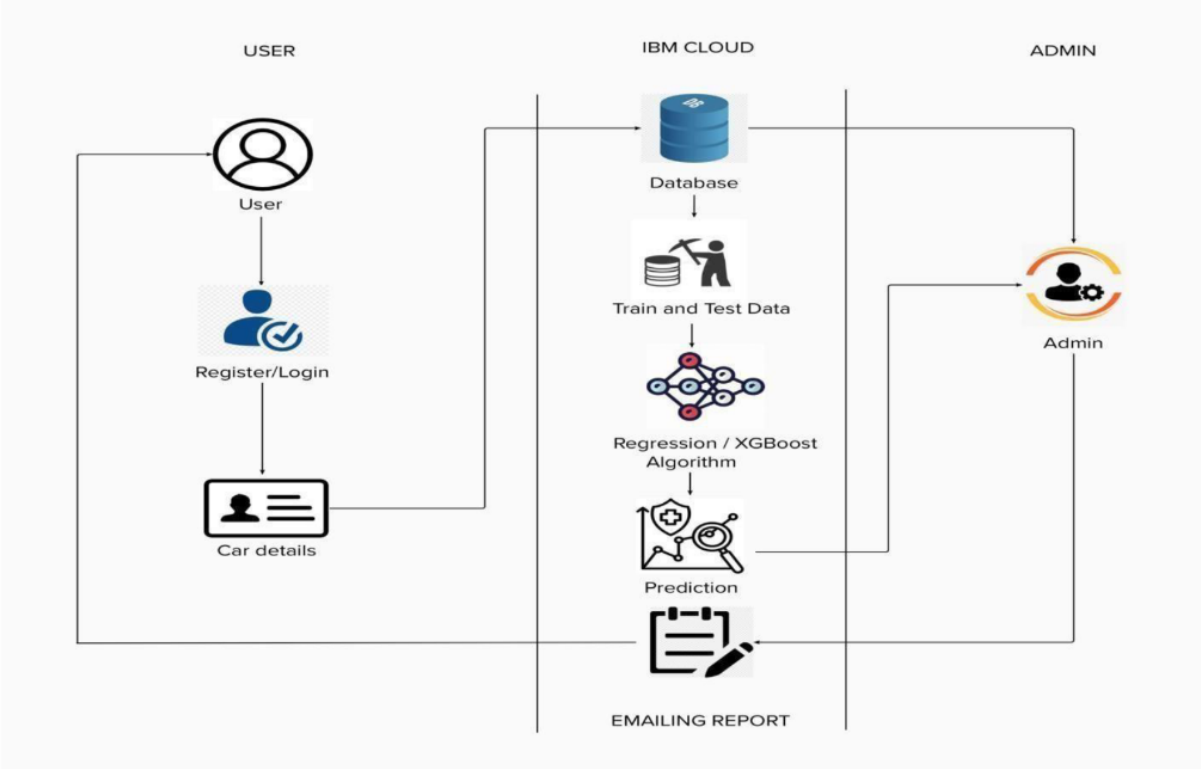


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	The user interacts with the application through Web UI.	HTML, CSS, JavaScript, Flask - Python framework, etc.
2.	Application Logic-1	Before enjoying the functionalities provided by the website, the user needs to register. Registration is meant to be done initially. Once an account is created, the login can be made using username and password	Flask
3.	Application Logic-2	After logging in, users can provide all the necessary details that are involved in predicting the resale value of the car. The details will be given to the trained ML model for making predictions. The predicted output will be displayed to the user.	Regression model (Machine learning
4.	Application Logic-3	This allows the users to enter their feedback. The provided feedback will be stored in the databases. This database helps in listing all the feedback provided by the user.	Database, Python, Flask
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	Machine Learning Model	To Improve the predictive accuracy and control over fitting	Random Forest Regressor Python
9.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Heroku Platform

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	To establish an connection between the flask and Python Flask an HTML page..	Python Flask
2.	Security Implementations	To Protect the user information as well as their car SHA-256, Encryptions details.	SHA-256, Encryptions
3.	Scalable Architecture	The model can be viewed and accessed in both computer as well as mobile phone.	Web UI, Mobile Android app
4.	Availability	The model can be available anywhere at any time	IBM Cloud
5.	Performance	The model performance has high accuracy and with portable from one machine to another machine	HTML, CSS

3. User Stories

User Stories

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

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	User can register for the application by entering username, email, password and confirming password, phone number.	I can get an idea about how car resale works	Low	Sprint-3
		USN-2	User will receive confirmation email once registered for the application.	Verify the registered account.	High	Sprint-1
		USN-3	Validation of the user can be done directly using Gmail.	Account validated and got access to dashboard	Low	Sprint-2
	Login	USN-4	Enter the Username and Password to login to the application.	Right account credentials should be entered	Medium	Sprint-1
	Dasboard	USN-5	User can view the detail about the website	.	High	Sprint-2
		USN-6	User can view the account details and history	User should be verified	Medium	Sprint-2
Customer (Web user)		USN-7	User can give the feedback on the accuracy of the prediction and on the user interface		High	Sprint-2
Core development team	Core function	USN-8	Design Develop the application in the way of the best user interface and maintenance should be take care	Easy and selfunderstandable user interface	High	Sprint-2
Administrator		USN-9	The website should be responsive on all the device and the screen size	User experience should be good responsive of the device	Medium	Sprint-3
		USN-10	Collect the dataset and process the data		High	Sprint-3
User in website	Enter the detail about car	USN-8	User should enter the information about the car.	Entered data should be correct	High	Sprint-4
		USN-11	User can view the prediction value		High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

1. Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset	USN-1	Downloading the dataset	5	High	Akash G Abhishek VS Aravind Karti S Harshini R
Sprint-1		USN-2	Visualizing the dataset	2	Low	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-1		USN-3	Pre-process the dataset	3	Medium	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-2	User Interface	USN-4	Random Forest Regressor model building	5	High	Akash G Abhishek V S Aravind Karti S Harshini R

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-2		USN-5	Model Integration with flask	5	High	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-2		USN-6	Build HTML Pages	3	Medium	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-3	Required inputs from User	USN-7	Dashboard accessibility	5	High	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-3		USN-8	Select the factors of car	1	Low	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-3		USN-9	Required factors are filled and car resale value is predicted	5	High	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-4	Deploy the website	USN-10	Register on IBM Cloud	1	Low	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-4		USN-1	Train the ML model on IBM Cloud	5	Medium	Akash G Abhishek V S Aravind Karti S Harshini R
Sprint-4		USN-12	Deploy the website on IBM Cloud	5	High	Akash G Abhishek V S Aravind Karti S Harshini R

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	10	6 Days	25 Oct 2022	30 Oct 2022	10	30 Oct 2022
Sprint-2	13	6 Days	01 Nov 2022	06 Nov 2022	13	06 Nov 2022
Sprint-3	11	6 Days	08 Nov 2022	13 Nov 2022	11	13 Nov 2022
Sprint-4	11	6 Days	15 Nov 2022	20 Nov 2022	11	20 Nov 2022

3. Reports from JIRA

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

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

VELOCITY OF THE PROJECT – CAR RESALE VALUE PREDICTION

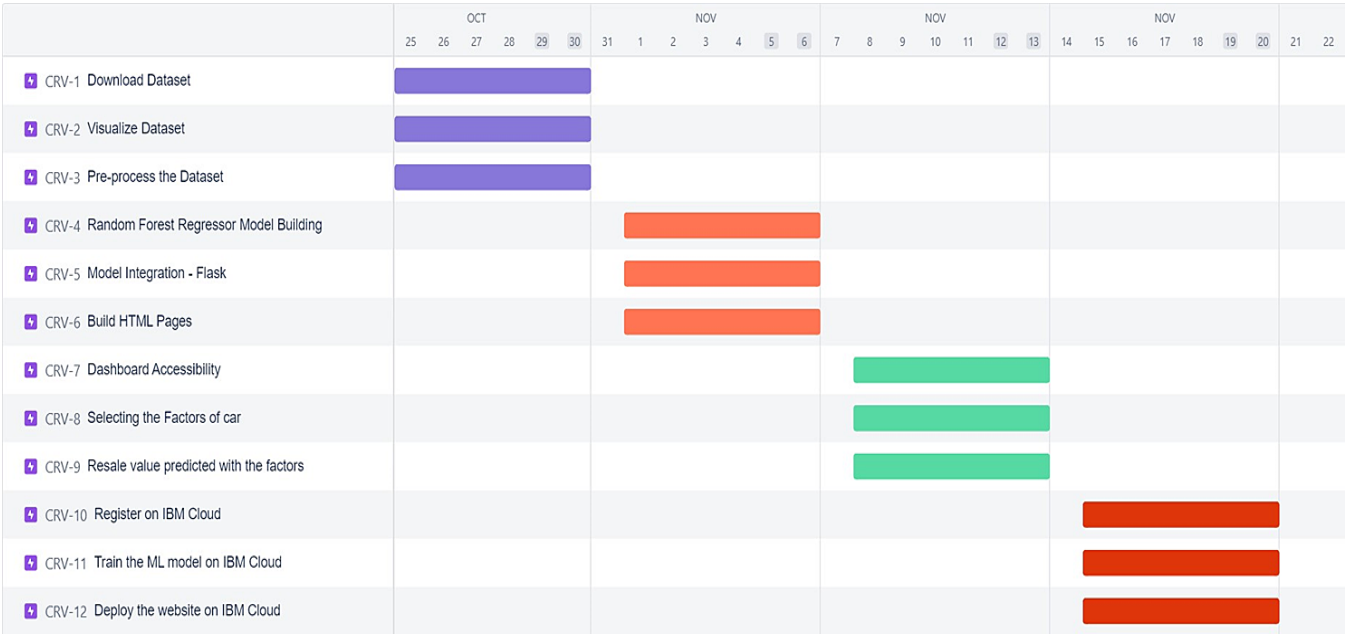
Sprint-1 = 10/6 = 1.66

Sprint-2 = 13/6 = 2.16

Sprint-3 = 11/6 = 1.83

Sprint-4 = 11/6 = 1.83

Total Velocity = 12.2 /4 = 1.87


OCT
NOV
NOV
NOV
25 26 27 28 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22
CRV-1 Download Dataset
CRV-2 Visualize Dataset
CRV-3 Pre-process the Dataset
CRV-4 Random Forest Regressor Model Building
CRV-5 Model Integration - Flask
CRV-6 Build HTML Pages
CRV-7 Dashboard Accessibility
CRV-8 Selecting the Factors of car
CRV-9 Resale value predicted with the factors
CRV-10 Register on IBM Cloud
CRV-11 Train the ML model on IBM Cloud
CRV-12 Deploy the website on IBM Cloud

7. CODING & SOLUTIONING

1. Feature 1

```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="utf-8">
     <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
    <link rel="stylesheet" href="style.css">
    <title>Flight Delay Prediction</title>
    <style>
       body {
    font-family: sans-serif;
    /* background-image: url(images\ \(1\).jfif); */
    background-image: url(https://wallpaperaccess.com/full/42422.jpg);
    background-repeat: no-repeat;
    /* overflow: hidden; */
    background-size: cover;
    color: wheat;
}

.container {
    width: 50%;
    margin: 3% auto;
    border-radius: 25px;
    background-color: rgba(0,0,0,0.1);
    box-shadow: 0 0 15px #333;
}

.header {
    text-align: center;
    padding-top: 15px;
    padding-bottom: 5px;
```

```
}

.header h1 {
    color: #333;
    font-size: 40px;
    margin-bottom: 5px;
}

.main {
    text-align: center;

}

td {
    color:black;
    font-weight: bold;
    font-size: 20px;
    padding: 5px 80px 5px 90px;
}

input[type=number],select {
    width: 100%;
    padding: 10px 16px;
    margin: 6px 0;
    display: inline-block;
    border: 1px solid #ccc;
    border-radius: 4px;
    box-sizing: border-box;
    background-color: rgb(224, 245, 231);
    position: relative;
  }

label {
    font-size: 20px;
    color: #333;
```

```
    text-align: left;
    text-align: center;
    position: relative;
}

input[type=submit] {
    width: 60%;
    background-color: #676767;
    color: white;
    font-size: 20px;
    padding: 10px 15px;
    margin: 8px 0;
    border: none;
    border-radius: 4px;
    cursor: pointer;
  }

  input[type=submit]:hover {
    /* background-color: #703f36;; */
    box-shadow: 2px 2px 5px rgb(104, 104, 104);
    background-color:  rgb(69, 69, 69);
    border-radius: 4px;
  }

  input[type=number]:hover {
    /* background-color: #703f36;; */
    box-shadow: 2px 2px 5px rgb(163, 163, 163);
    background-color:  rgb(195, 199, 197);
    border-radius: 4px;
  }

    </style>
</head>

<body>
```

```
<div class="container">

                    <div class="header"><h1>CAR RESALE VALUE PREDICTION</h1><br></div>
    <div class="main">
    <form action="{{ url_for('Predict')}}" method="post">


      <table>
 <b>
     <tr style="text-align: left;"><td>Abtest :</td>  <td><input type="number" name="abtest" placeholder="Enter the abtest" required="required"></td></tr>
   <tr style="text-align: left;"><td>VehicleType :</td>  <td><input type="number" name="vehicleType" placeholder="Enter the vehicleType" required="required"></td></tr>
      <tr style="text-align: left;"><td>yearOfRegistration :</td>  <td><input type="number" name="yearOfRegistration" placeholder="Enter the yearOfRegistration" required="required"></td></tr>
     <tr style="text-align: left;"><td>gearbox :</td>  <td><input type="number" name="gearbox" placeholder="Enter the gearbox" required="required"></td></tr>
     <tr style="text-align: left;"><td>powerPS :</td>  <td><input type="number" name="powerPS" placeholder="Enter the powerPS " required="required"></td></tr>
     <tr style="text-align: left;"><td>model :</td>  <td> <input type="number" name="model" placeholder="Enter the model" required="required"></td></tr>
    <tr style="text-align: left;"><td>kilometer :</td>  <td><input type="number" name="kilometer" placeholder="Enter the kilometer" required="required"></td></tr>
      <tr style="text-align: left;"><td>monthOfRegistration:</td>  <td><input type="number" name="monthOfRegistration" placeholder="Enter the monthOfRegistration" required="required"></td></tr>
     <tr style="text-align: left;"><td>fuelType:</td>  <td> <input type="number" name="fuelType" placeholder="Enter the fuelType" required="required"></td></tr>
      <tr style="text-align: left;"><td>brand:</td>  <td> <input type="number" name="brand" placeholder="Enter the brand" required="required"></td></tr>
      <tr style="text-align: left;"><td>notRepairedDamage:</td>  <td><input
```

```
type="number" name="notRepairedDamage" placeholder="Enter the notRepairedDamage" required="required"></td></tr>
      <tr style="text-align: left;"><td>postalCode:</td> <td><input type="number" name="postalCode" placeholder="Enter the postalCode" required="required"></td></tr>


        </b>
         </table>
         <input type="submit" , value='PREDICT!' >



      </form>
      </div>

   </div>



</body>

</html>
```

2. Feature 2

```
<!DOCTYPE html>
<html lang="en">
 <head>
   <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
   <title>Flight Delay Prediction</title>
   <style>
      body {
   font-family: sans-serif;
   /* background-image: url(images\ \(1\).jfif); */
```

```
                                                    background-image:
url(https://www.hdcarwallpapers.com/thumbs/2022/koenigsegg_cc850_2022_4k_8k_
2-t2.jpg);
    background-repeat: no-repeat;
    overflow: hidden;
    background-size: cover;
    color: wheat;
}
.center {
  margin: auto;
  width: 60%;
  padding: 50px;
}

    </style>
                                                    <link          rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"
integrity="sha384-
9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj
7Sk" crossorigin="anonymous">
  </head>

<body style="background-color: rgb(214, 197, 160);">

        <center>

            <h1>  PREDICTION :  </h1>
            <div class="center">
        <h2 style="color:green;font-weight:bolder;" >₹{{ data }}</h2>
            <br><br></div>
        <a href='/' style="font-size:20px;color: purple;">Go back to home page</a>
        </center>

</body>
</html>
```

7.3 Training the Model:

```
import pandas as pd
import numpy as np
import pickle
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import sklearn
from sklearn.preprocessing import LabelEncoder
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import GradientBoostingClassifier, RandomForestClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import RandomizedSearchCV
import imblearn
from imblearn.under_sampling import RandomUnderSampler
from sklearn.preprocessing import scale
from sklearn.metrics import mean_squared_error
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix,
f1_score
import warnings
warnings.filterwarnings('ignore')
```

Reading the Dataset:

```
df = pd.read_csv('autos.csv', parse_dates=['dateCrawled', 'dateCreated', 'lastSeen'],
encoding = 'latin1')
df.head()
df.info()
df.shape
```

Cleaning the Dataset:

```
df.columns
# Rearranging the Columns
df = df[['dateCrawled', 'name', 'seller', 'offerType', 'abtest',
      'vehicleType', 'yearOfRegistration', 'gearbox', 'powerPS', 'model',
      'kilometer', 'monthOfRegistration', 'fuelType', 'brand',
      'notRepairedDamage', 'dateCreated', 'nrOfPictures', 'postalCode',
      'lastSeen', 'price']]
```

```
# Droping the Unwanted Columns
df.drop(columns= ['seller', 'offerType', 'nrOfPictures'], inplace = True)
df.drop(columns= ['name','dateCrawled', 'dateCreated', 'lastSeen'], inplace = True)
```

Identifying and Removing Missing Values:

```
# Checking for Missing Values
df.isna().sum()
# Removing Missing Values
df['vehicleType'].fillna(df['vehicleType'].mode()[0], inplace = True)
df['gearbox'].fillna(df['gearbox'].mode()[0], inplace = True)
df['model'].fillna(df['model'].mode()[0], inplace = True)
df['fuelType'].fillna(df['fuelType'].mode()[0], inplace = True)
df['notRepairedDamage'].fillna(df['notRepairedDamage'].mode()[0], inplace = True)
df.isna().sum()
```

Check and Remove Duplicate Values:

```
# Checking for Duplicates
df.duplicated().sum()
# Removing Duplicates
df = df.drop_duplicates()
df.duplicated().sum()
```

Label Encoding:

```
df.info()
from sklearn.preprocessing import LabelEncoder
le = LabelEncoder()
df['abtest'] = le.fit_transform(df['abtest'])
df['vehicleType'] = le.fit_transform(df['vehicleType'])
df['gearbox'] = le.fit_transform(df['gearbox'])
df['model'] = le.fit_transform(df['model'])
df['fuelType'] = le.fit_transform(df['fuelType'])
df['brand'] = le.fit_transform(df['brand'])
df['notRepairedDamage'] = df['notRepairedDamage'].replace({'nein' : 0, 'ja' : 1})
```

```
df.info()
```

Identifying and Handling Outliers:

```
df.skew()
df.yearOfRegistration=df.yearOfRegistration.clip(lower=df.yearOfRegistration.quantile(0.05),upper=df.yearOfRegistration.quantile(0.95))
df.price=df.price.clip(lower=df.price.quantile(0.05),upper=df.price.quantile(0.95))
df.powerPS=df.powerPS.clip(lower=df.powerPS.quantile(0.05),upper=df.powerPS.quantile(0.95))
df.skew()
```

Visualization:

```
plt.figure(figsize=(20,20))
sns.heatmap(df.corr(), annot = True)
plt.show()
sns.pairplot(df)
plt.show()
```

Descriptive Statistics:

```
df.nunique()
df.describe()
df.skew()
df.kurt()
```

Splitting the Data:

```
# Splitting x and y variables
x = df.drop(columns = 'price')
y = df['price']
# Splitting into test and train
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=0)
```

7.4 Model Building:

Linear Regression

```
from sklearn.linear_model import LinearRegression
lr = LinearRegression()
lr.fit(x_train, y_train)
```

Lasso Regression

```
from sklearn.linear_model import Lasso
lasso = Lasso(alpha=0.01, normalize=True)
lasso.fit(x_train, y_train)
```

Decision Tree

```
from sklearn.tree import DecisionTreeRegressor
DT = DecisionTreeRegressor()
DT.fit(x_train, y_train)
```

KNN

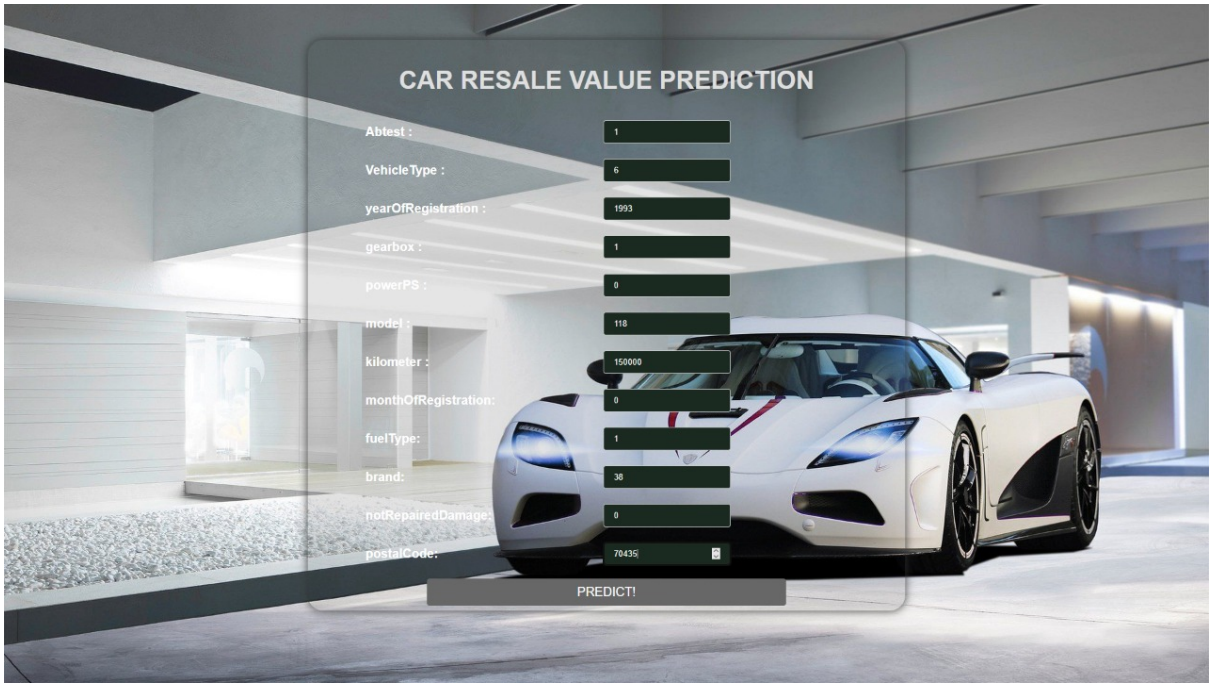
```
from sklearn.neighbors import KNeighborsRegressor
knn = KNeighborsRegressor()
knn.fit(x_train, y_train)
```

Random Forest

```
from sklearn.ensemble import RandomForestRegressor
RF = RandomForestRegressor()
RF.fit(x_train, y_train)
```

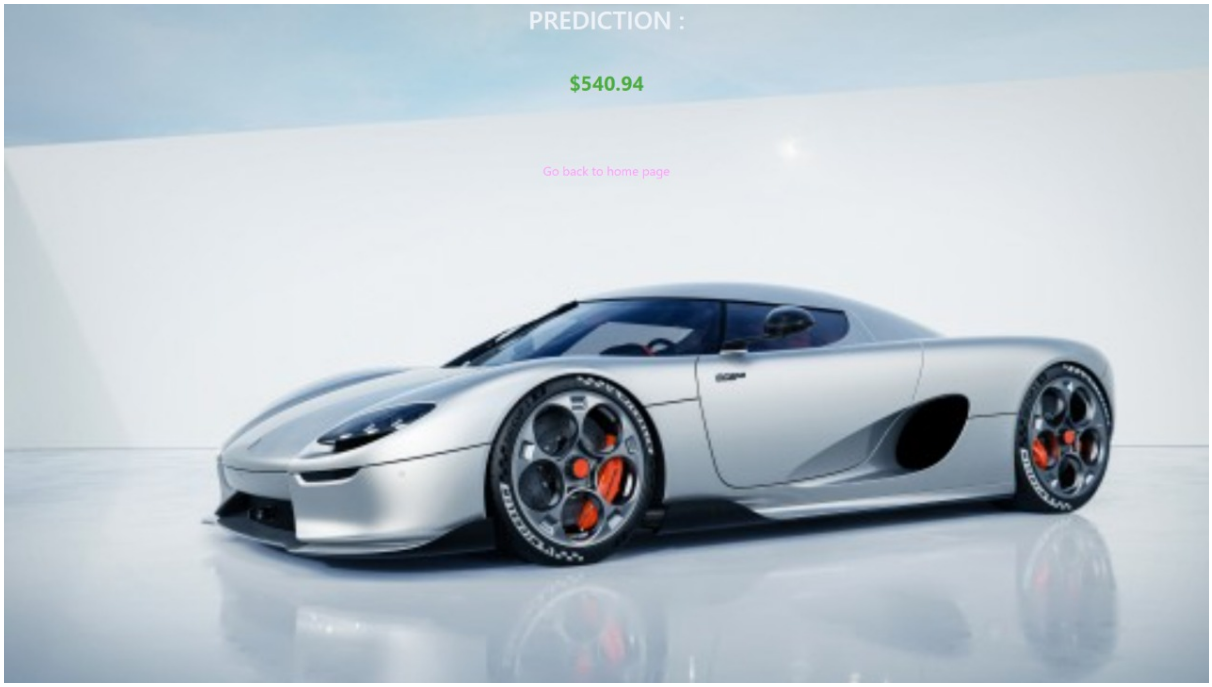
8. TESTING

1. Test Cases & User Acceptance Testing



9. RESULTS

1. Results



2. **Performance Metrics**

Checking the Metrics of the models

```
# Linear Regression

lr.score(x_test, y_test)

from sklearn.metrics import mean_squared_error

np.sqrt(mean_squared_error(y_test,lr.predict(x_test)))

# Lasso Regression

lasso.score(x_test, y_test)

np.sqrt(mean_squared_error(y_test,lasso.predict(x_test)))

# K Nearest Neighbour

knn.score(x_test, y_test)

np.sqrt(mean_squared_error(y_test,knn.predict(x_test)))

# Decision Tree

DT.score(x_test, y_test)

np.sqrt(mean_squared_error(y_test,DT.predict(x_test)))

# Random Forest

RF.score(x_test, y_test)

np.sqrt(mean_squared_error(y_test,RF.predict(x_test)))
```

3. **Evaluating the metrics**

Random Forest Model is Selected

```
RF.score(x_test, y_test)
```

Saving The Model

```
pickle.dump(RF, open('MODEL.pkl', 'wb'))
```

10. ADVANTAGES & DISADVANTAGES

Advantages:

- Effective in understanding non-linear, complex relationships
- Exceedingly interpretable and explicable
- Resilient to outliers
- Scaling of features is not essential.

Disadvantages:

- Requires more time.
- Requires a lot of computational power

11. CONCLUSION

It started by comprehending the use of machine learning in the automotive sector and how it has altered the driving experience. Next, an exploratory data analysis was conducted to examine the many variables that impact a used car's resale value (EDA). In addition, a Random Forest Regression model was created to forecast a used car's resale value. Finally, with the use of R squared score and Residual Plot to assess the model's performance. It also consists of the option of using less complex regression techniques, such as Lasso Regression and Linear Regression. However, before using them, it must ensure that the dataset is clear of outliers. The outliers are simpler to identify using pair plots and scatter plots.

12. FUTURE SCOPE

Future integration of this machine learning model with different websites that may supply real-time data for price prediction is possible. Additionally, we might include a lot of historical data on automobile prices, which would increase the machine learning model's accuracy. An Android app can be created as the user interface for interacting with users.

APPENDIX

Source Code

https://github.com/IBM-EPBL/IBM-Project-33829-1660227708/blob/main/DEVELOPMENT%20PHASE/SPRINT%20-%202/MODEL_BUILDING/EVALUATING%20AND%20SAVING%20THE%20MODEL.ipynb

Demo link:

https://www.youtube.com/embed/bAn0qUralY0

GitHub:

https://github.com/IBM-EPBL/IBM-Project-33829-1660227708