CAR RESALE VALUE PREDICTION

Team ID : PNT2022TMID10672

Team Leader : Iswarya.R

Team Members : Kanimozhi.V

Madhumidha.R

Pavithra.R

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION:

In this project, we have developed car resale value prediction systems using a variety of algorithms and methodologies that take into account a variety of car. In a word, automobile resale value prediction enables users to forecast the resale value of a car based on features such as miles driven, fuel type, etc. The goal of a car resale value prediction system is to forecast the accurate worth of used cars, enabling customers to sell their vehicles remotely with unbiased valuation and without the need for human participation. There is a critical need to close this gap between sellers and buyers due to the enormous demand for used cars and the shortage of professionals who can evaluate the proper valuation. The goal of this research is to create a system that can impartially forecast a car's resale value based on little information such as the number of miles travelled and the year of purchase.

1.2 PROJECT OVERVIEW:

It might be challenging to tell if the listed price is true due to the many factors that affect a used car's market value. In order to enable purchasers to make informed choices, our project aims to develop machine learning models that can accurately anticipate the price of a used car based on its characteristics. We construct and evaluate a number of learning methods using a dataset that contains the sale prices of various brands and models. The cost of the car will be established based on a variety of variables. Because they provide us with a continuous number as an output rather than a classified value, multiple regression algorithms enable us to predict the actual cost of an automobile rather than just its price range. Additionally, a user interface that accepts input from any user and displays the pricing of a car based on their inputs has been developed. Here, there are three different fuel data set types diesel, gasoline, and LPG are all used. For estimating car prices, we employ the K-Nearest Neighbor method, Decision Tree Classifier, Forest Regression, and Support Vector Machine technique.

1.3 PURPOSE:

The project aims to build a model that can predict and perform as a higher rate of car resale value based on the various attributes. The model would help in decreasing the risk activities and complexity involved in purchasing a valuable car through the prediction of many studies rather than the involvement of broker-dealers. It is used to forecast the proper valuation of used cars, allowing customers to sell the car remotely with perfect valuation and without human interaction in the process to avoid biased appraisal. Understanding the problem and establishing if it is a statistical technique or a predictor variable, as well as determining the prior data using various data pre-processing techniques. Obviously the employment of

various algorithms is based on the data collection. Create web applications with the flask framework to predict over car resale value.

CHAPTER -2

LITERATURE SURVEY

TOPIC 1: Used Car Price Prediction and Valuation Using Data Mining Techniques

AUTHOR: Abdulla

DESCRIPTION: Data mining technologies was used to predict car pricing. Three regressors (Random Forest Regressor, Linear Regression and Bagging Regressor were trained and evaluated against benchmark dataset.

TOPIC 2: Predicting the price of Second-hand Cars using Artificial Neural Networks

AUTHOR: Saamiayah Peerum, Nushrah Henna Chummun

DESCRIPTION:The purpose of this research is to determine whether artificial neural networks can forecast the price of used vehicles.

TOPIC 3: Car Price Prediction Using Machine Learning Techniques

AUTHOR: Enis Gegic, Becir Isakovic

DESCRIPTION: To create the predictions. several approaches such as multiple linear regression analysis, k- nearest neighbours, naive bayes, and decision trees were applied.

TOPIC 4: Used car Price Prediction Using Supervised Learning

AUTHOR: Venkatasubbu, Ganesh

DESCRIPTION: By use of machine learning algorithms such as Lasso Regression, Multiple Regression, and Regression Trees to develop a statistical model that can predict the price of a used car based on previous consumer data and a given set of features, as well as compare the prediction accuracy of these models to determine the best.

2.1 EXISTING SYSTEM:

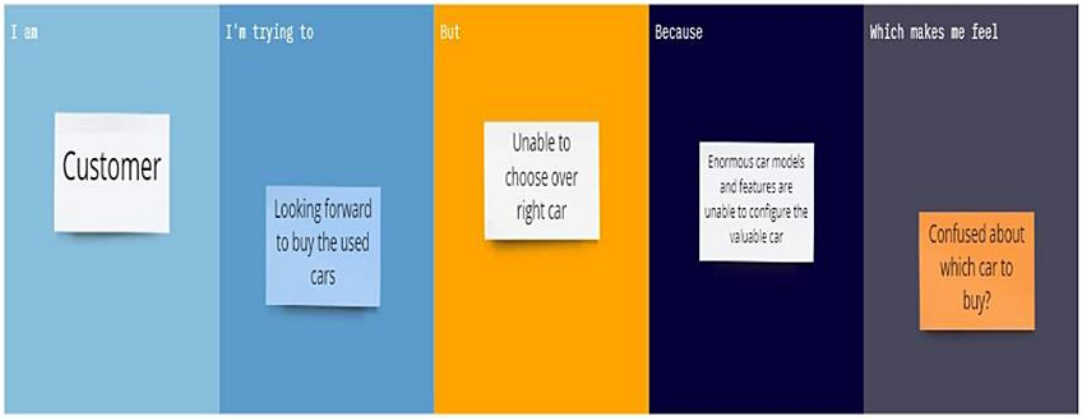
As more people are opting to buy used cars, resale values are rising significantly. As a result, there is an urgent need for technology that can estimate used automobile prices properly assesses the value of the car using a range of factors. The current method uses a tactic where a vendor picks a price at random without disclosing the car's current market value to the buyer. In reality, though, neither the seller nor the price at which he should sell the car are aware of its current market value.In difficult economic circumstances, it is possible that sales of used automobiles and used imported cars would increase in many developed economies, renting a car is more common than buying one entirely.

2.2 REFERENCES:

1. N. Monburinon, P. Chertchom, T. Kaewkiriya, S. Rungpheung, S. Buya and P. Boonpou, "Prediction of prices for used car by using regression models," 2018 5th International Conference on Business and Industrial Research (ICBIR), Bangkok, 2018, pp. 115-119.
2. RICHARDSON, M., 2009. Determinants of Used Car Resale Value. Thesis
3. (BSc). The Colorado College.
4. Sameerchand Pudaruth, “Predicting the Price of Used Cars using Machine Learning Techniques”;(IJICT 2014).
5. Enis gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric, ”Car Price Prediction Using Machine Learning”; (TEM Journal 2019)

6. Sabir Buya, Pitchayakit Boonpou, “Prediction of Prices for Used Car by using Regression Models” (ICBIR 2018).
7. Doan Van Thai, Luong Ngoc Son, Pham Vu Tien, Nguyen Nhat Anh, Nguyen Thi Ngoc Anh, “Prediction car prices using qualify qualitative data and knowledge-based system” (Hanoi National University)

2.3 PROBLEM STATEMENT DEFINITION:



Problem Statement (PS)	**I am (Customer)**	**I’m trying to**	**But**	**Because**	**Which makes me feel**
PS-1	Professiona l	Hoping to buy second hand cars	Analyzing the manufactur	Unaware of car related aspects	Confused

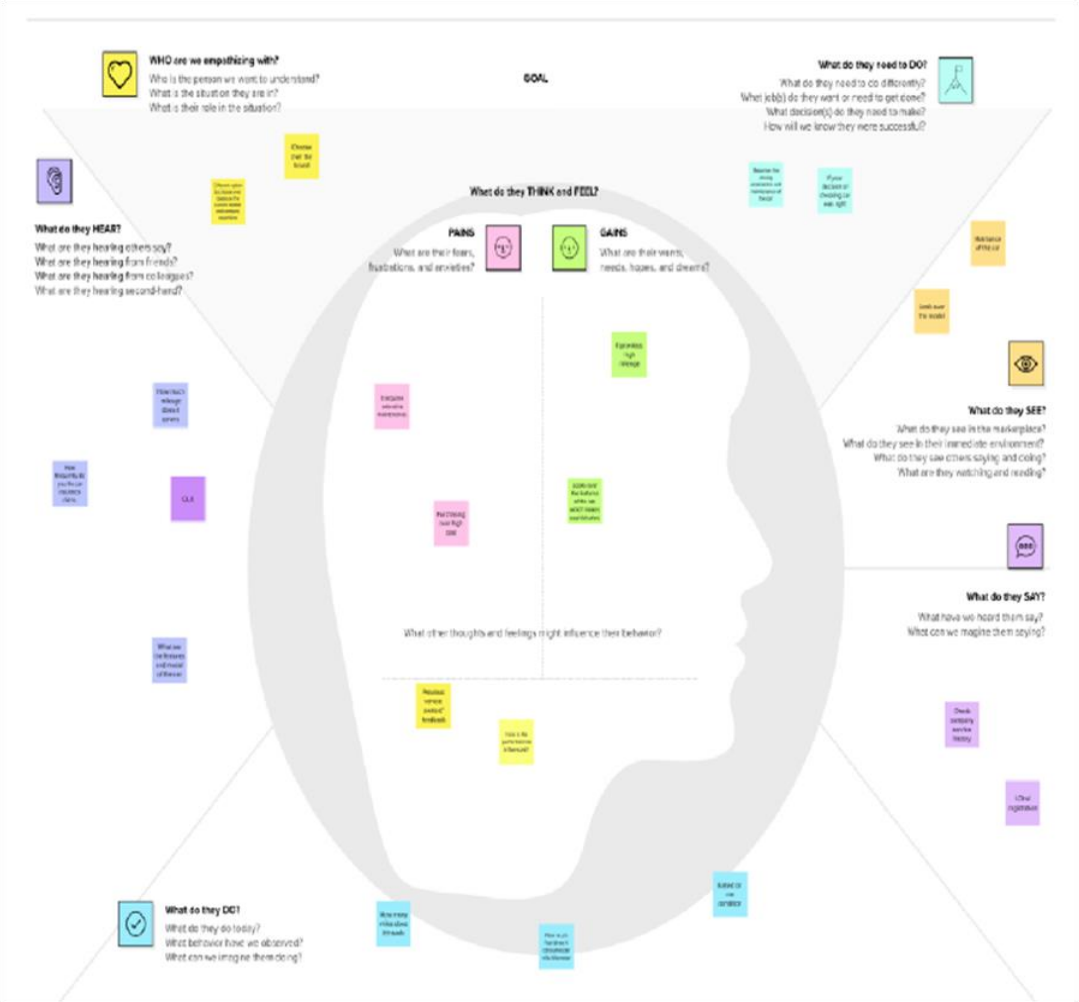
			ed year, brand models and pricing in the market price	difficult to forecast	
PS-2	Entrepreneurs	Looking over worthy used cars	Evaluating the various features and comforts of the car	Look finest and comfort to drive hard to predict	Complicate
PS-3	Normal People	Exploring the value for pricing of used cars	Examine the car's price and condition	The current resale prediction is tough to estimate	Distress
PS-4	Traveller	Desire for long-distance travel vehicles	Unable to predict the mileage covered by the different models	Don't have proper guidance	Intricate

CHAPTER-3

IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS:

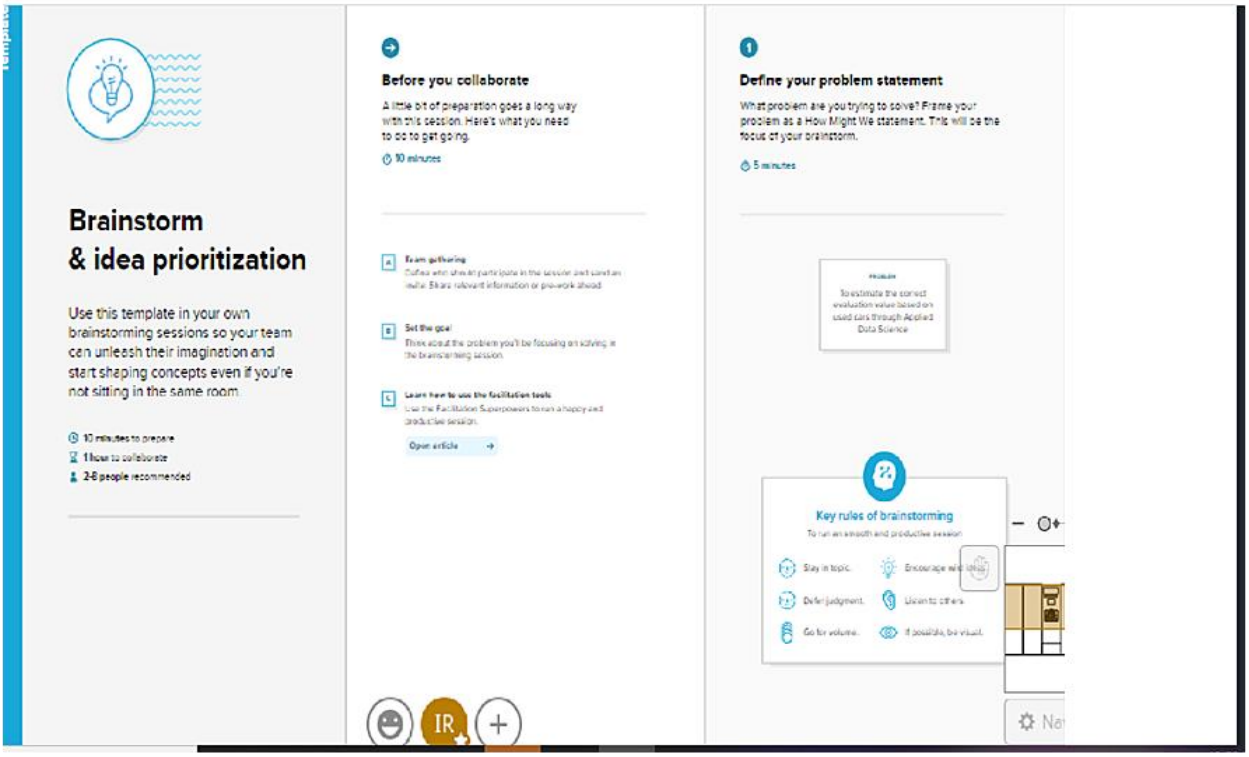
The Empathy Map Canvas helps teams develop deep, shared understanding and empathy for other people. People use it to help them improve customer experience, to navigate organizational politics, to design better work environments, and a host of other things



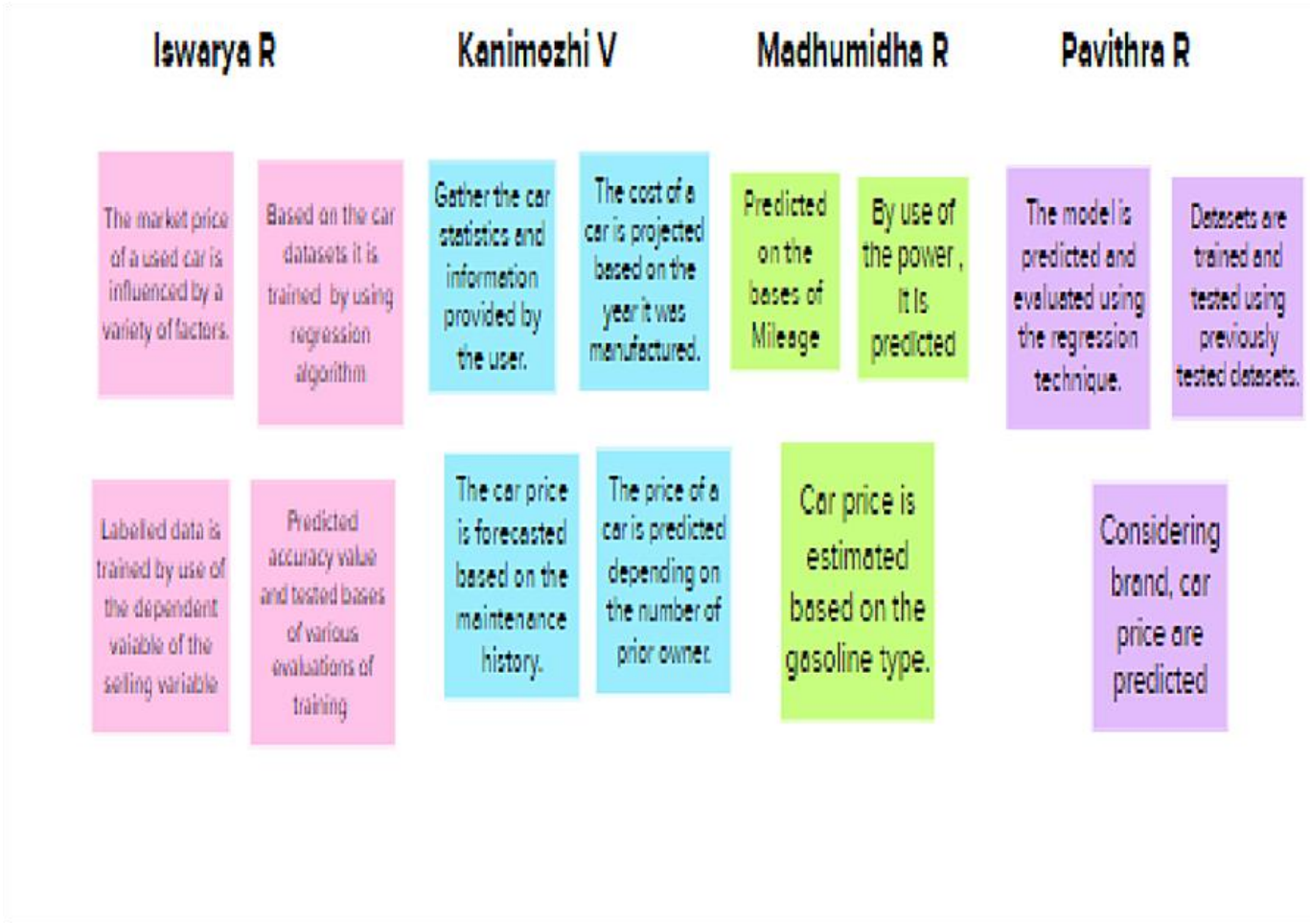
3.2 IDEATION & BRAINSTORMING:

Brainstroming is the term which breaks any idea that come to our mind which addresses our problem statement.It deals to discuss the ides to the team members and gather the ideas from them.Each team members reveals their ideas about bank loan prediction such as check the loan amount of the applicant, Occupation of the applicant, Gender of the applicant, Marital status of the applicant, Identity proof of the applicant.It involves to collect group ideas, Ideas prioritization.

Step-1: Team Gathering, Collaboration and Select the Problem Statement



Step-2: Brainstorm, Idea Listing and Grouping



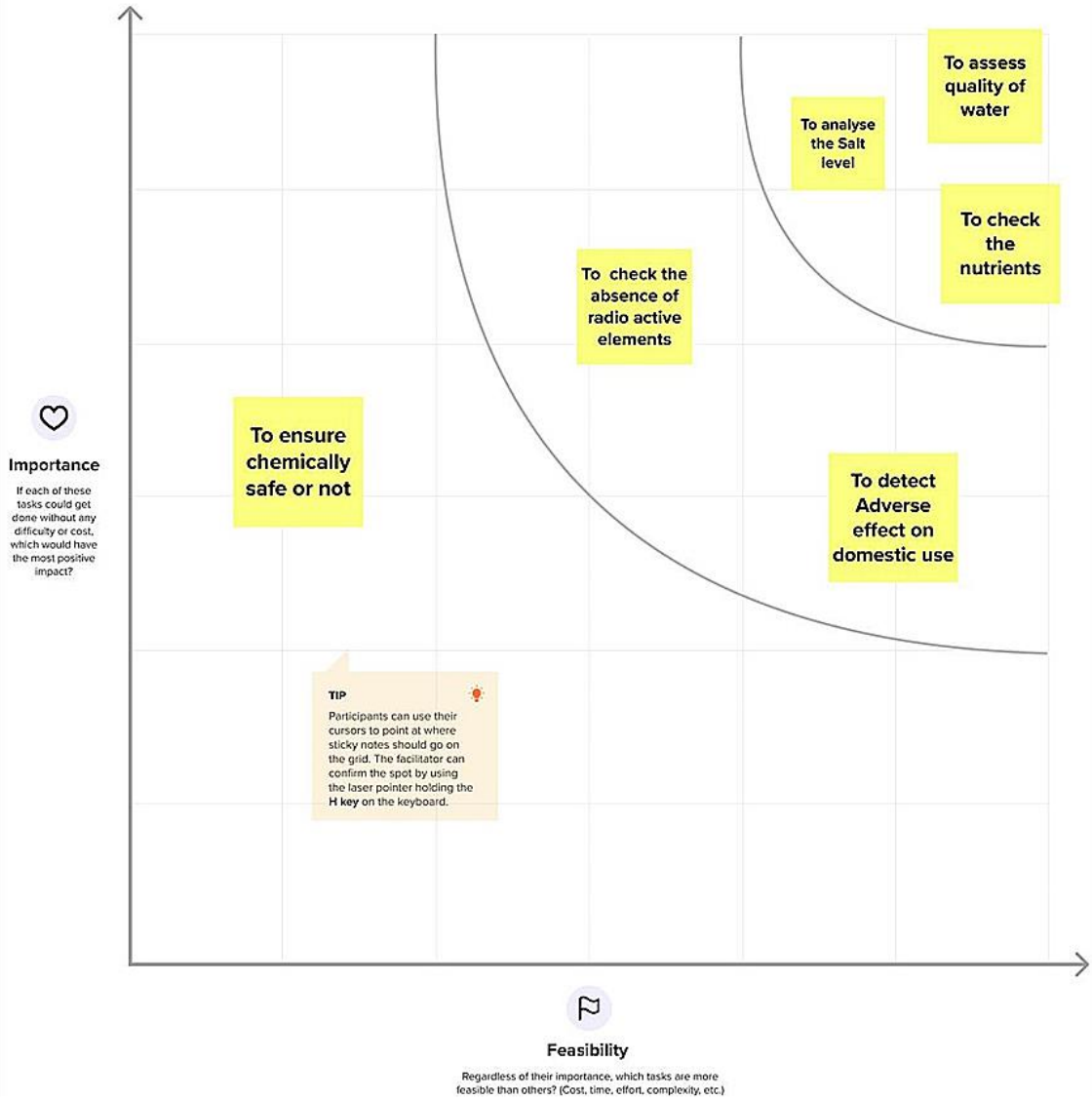
Step-3: Idea Prioritization

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

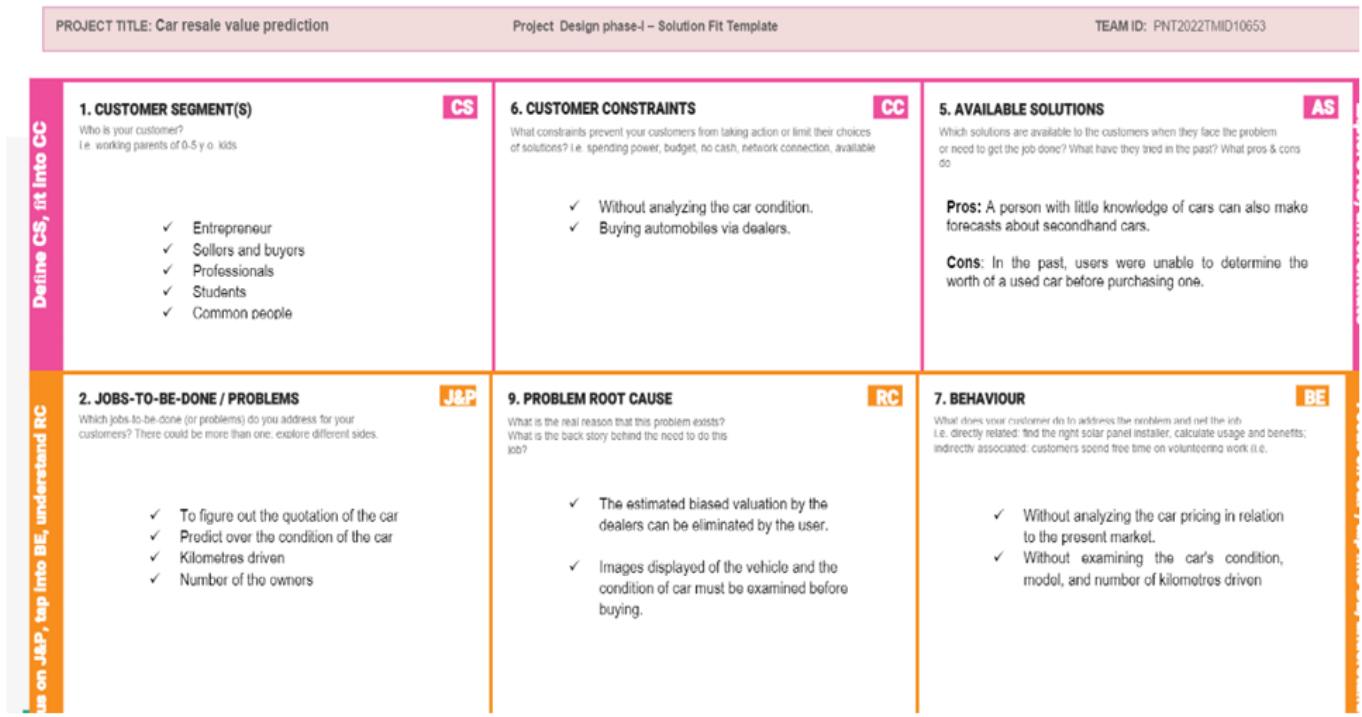


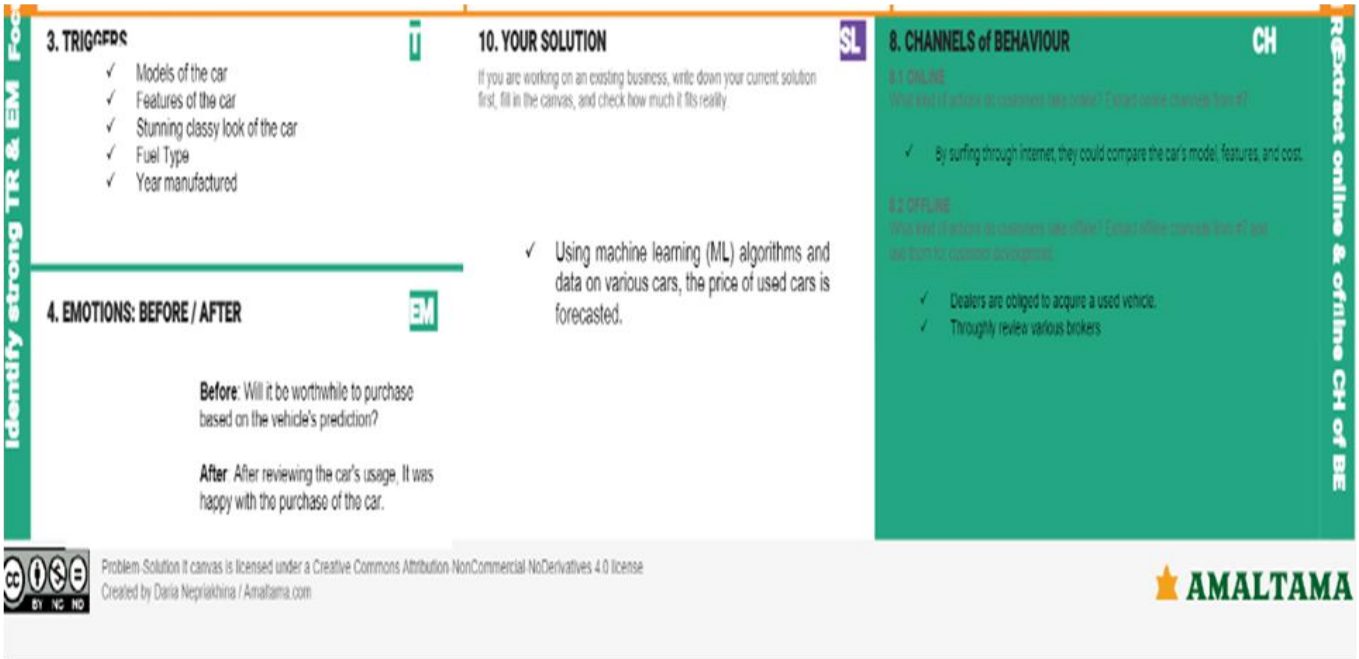
3.3 PROPOSED SOLUTION:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To estimate the resale value of a used and previously owned vehicle The value and model of a car are impacted by its use, mileage, and other other factors. It will be difficult to address this issue. To estimate the market value of any old car by incorporating each contributor's input
2.	Idea / Solution description	The accuracy score of the machine learning technique is employed in this process to predict the used car price.
3.	Novelty / Uniqueness	Used car price prediction uses information such as year, model, mileage(km),and others to swiftly determine the worth of a car and has a high accuracy rate of prediction.
4.	Social Impact / Customer Satisfaction	The user interface is indeed appealing and pragmatic. This system is available to all users, regardless of device compatibility. The worth of a car will also be determined by crucial social characteristics that

		provide the greatest user experience precise
5.	Business Model (Revenue Model)	It enables users to predict the correct valuation of a car price remotely with perfect valuation and without human intervention such as car dealers in the process to eliminate biased valuation predicted by the dealer.
6.	Scalability of the Solution	In order to produce a scalable solution, it contains a feature set with several factors that contribute to automobile price. As a result, even as the dataset expand, the solution will be able to make accurate predictions. Furthermore once the software is implemented, mobile users will be able to access the cloud

3.4 PROBLEM SOLUTION FIT:





CHAPTER -4

REQUIREMENT ANALYSIS

4.1 FUNCTIONALREQUIREMENTS:

Following are the functional requirements of the proposed solution.

FR No.	**Functional Requirement(Epic)**	**Sub Requirement (Story/Sub-Task)**
FR-1	User Registration	Using the Email id and setting the password
FR-2	User confirmation	Confirmation via Gmail Confirmation via OTP
FR-3	User Login	Using the registered Email id and password login
FR-4	Dashboard	It is viewing the details such as the profile page navigation

FR-5	Value prediction	It is the system to predict the amount of resale value based on the parameters provided by the user enters the details of the car into the from given and accordingly the car resale value is predictable
FR-6	Car Condition	To predict the value of a car factors such as usage mileage model and so on are considered
FR-7	Model of car availability	The user can search for and select a car from among the different car models available as well as compare it to other cars if desired
FR-8	Feedback	It get feedback on the entire end to end usage of the predictive systems while continuing to improve the accuracy prediction.

4.2 NON-FUNCTIONALREQUIREMENTS:
Following are the non-functional requirements of the proposed solution.

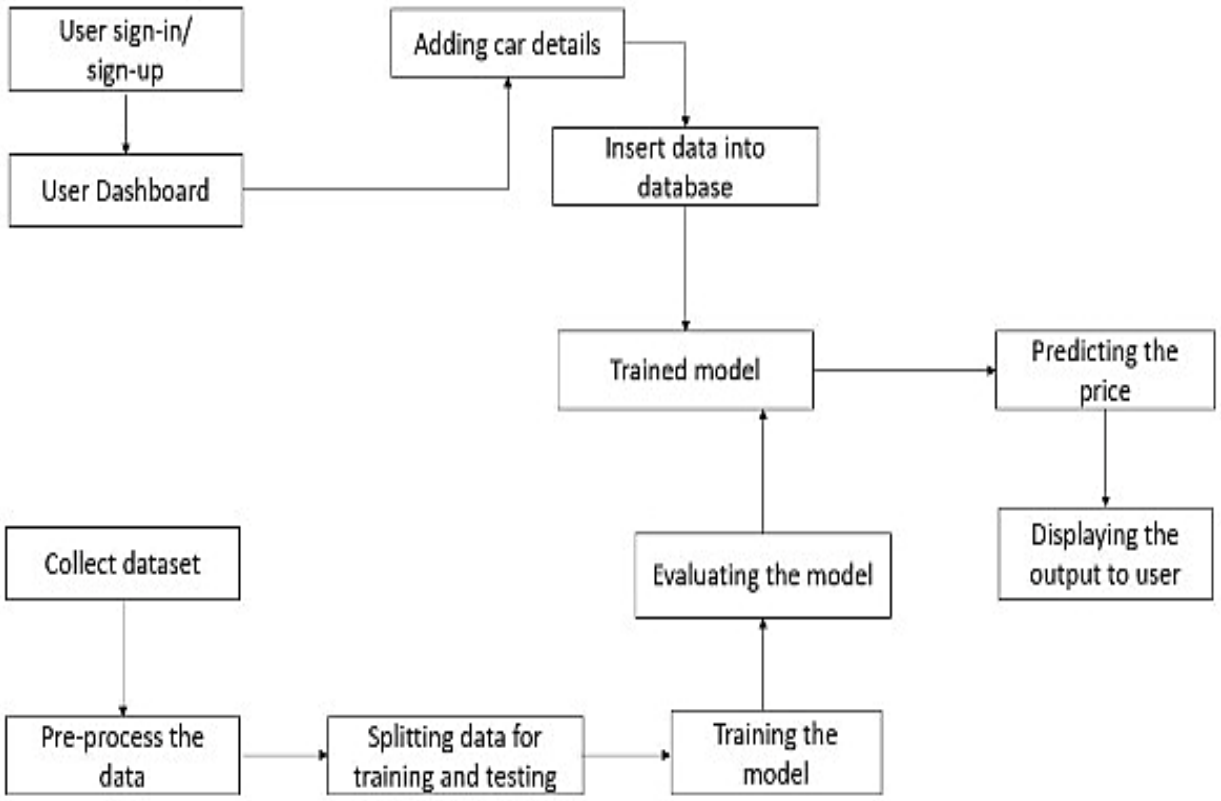
FR No.	Non-Functional Requirement	Description
NFR-1	Usability	In order to predict the resale value of the car, we proposed an intelligent, flexible, and effective system that is based on using regression algorithm Considering the main factors which would affect the resale value of a vehicle a regression model is to be built that would give the nearest resale value of the vehicle.
NFR-2	Security	Assures all data inside the system its part will be protected against malware attacks, unauthorized access. For instance such details should be considered under what circumstances the unauthorized access takes place,
NFR-3	Reliablity	The probability of a system is defined by it operation and maintenance, over a period of time, without making any proper changes. For

		the control system of a hybrid car that works on the vibrational and magnetic environment, high reliability must meet the executive and the security requirements of the entire system. Reliability of the substrate was noted from the various stages of designing, installation, and testing.
NFR-4	Performance	It has high performance and measure the model is performance and high accuracy by using the machine learning techniques.
NFR-5	Availability	It used Car price in the market Get instant payment of all types of car brand such as the Honda, Tata , Hyundai, Skoda, Toyota, Ford etc
NFR-6	Scalability	By testing different models, it was attempted to gain alternative insights and eventually compare their performance and car types

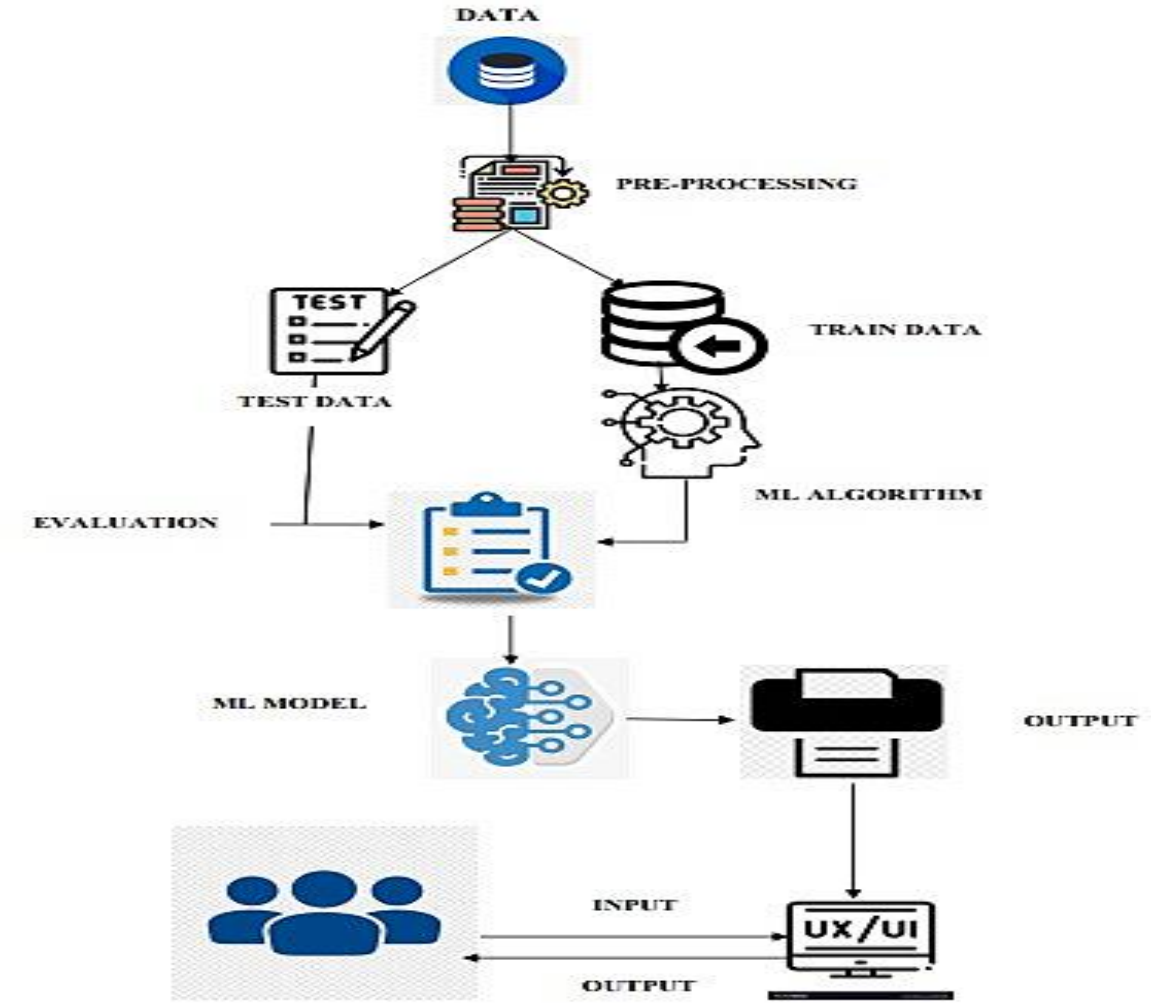
CHAPTER-5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS:



5.2 SOLUTION & TECHNICAL ARCHITECTURE:



5.3 USER STORIES:

User type	Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Priority
Customer(mobile user)	Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password and confirming my password	High
	Sprint-1		USN-2	As a user, I will receive confirmation email once I have registered for the application	High
	Sprint-2		USN-3	As a user, I can register for the application through Facebook	Low

	Sprint-1		USN-4	As a user, I can register for the application through Gmail	Medium
	Sprint-1	Login	USN-5	As a user, I can log into the application by entering email & password	High
	Sprint-2	Dashboard	USN-6	As a user, I can enter dashboard via my id	Medium
Customer(web user)	Sprint-1	Registration	USN-7	As a user, I can register in official website.	Medium
Customer care executive	Sprint-1		USN-8	As a user, I can't access my profile.	High
Administrator	Sprint-2		USM-9	Admin maintain the details of the car	High

CHAPTER-6

PROJECT PLANNING AND SCHEDULING

6.1. SPRINT PLANNING &ESTIMATION:

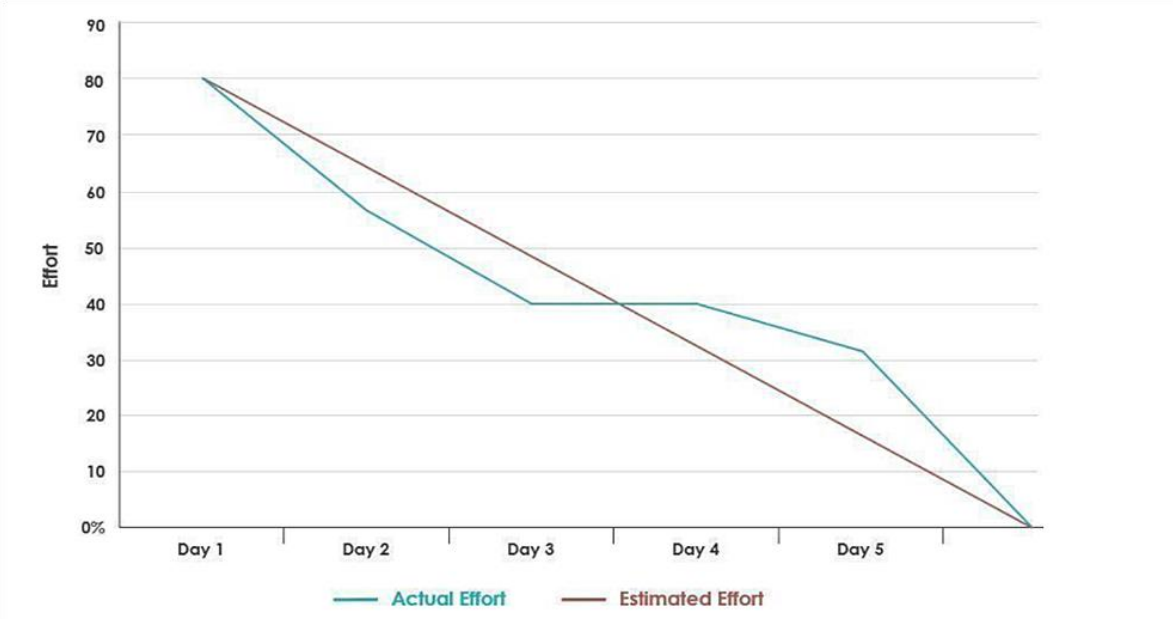
Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story Points**	**Priority**	**Team Members**
Sprint-1	Dataset reading and Data Preprocessing	USN-1	Cleaning the dataset and splitting to dependent and independent values	2	High	Iswarya R
Sprint-2	Building the model		Handle all the missing values in the dataset			Kanimozhi v Iswarya R
Sprint-3	Application Building	USN-3	Calculate the water quality index using the collected dataset	2	Medium	Pavithra R Madhumidha R

Sprint-4	Train the model in the IBM cloud	USN -4	Visualize the data using the histogram and heatmaps.	2	Medium	Iswarya R Kanimozhi V

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	5 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	5 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	5 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	5 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

6.3 REPORTS FROM JIRA:



CHAPTER-7

CODING & SOLUTIONING

FEATURE :

Due of a lack of data, the system can only handle cars at the moment. Additionally, information has been gathered that can be used to a variety of automobile types cities to increase usability and accuracy. Once sufficient data has been gathered, efficient deep learning techniques like the Random Forest, KNN algorithm, and multilinear regression can be used. This can increase precision and reduce only a few features are currently used to forecast a car's resale value. Additional features could be added to this physical condition of the car can also be inferred from photos, such as by spotting dents, scratches, etc., leading to a more accurate estimation of the car's resale worth.

Random forest algorithm:

```
def randomForest(x_train,x_test,y_train,y_test):
    rf = RandomForestClassifier()
    rf.fit(x_train,y_train)
    pred_test = rf.predict(x_test)
    print('Confusion Matrix')
    print(confusion_matrix(y_test,pred_test))
    print('Classification Report')
   print(classification_report(y_test,pred_test))
   print('Score')
   print(rf.score(x_test,y_test))
```

Comparison of Random Forest Algorithm Vs KNN Vs Multilinear regression:

Numerous algorithms, including KNN, multilinear regression, and random forests. The discussion mainly concentrated on the statistical and mathematical features of each algorithm applicability for particular use situations as well as their key shortcomings. In the domains of machine learning and data science, the endeavour we are doing to summarise the numerous algorithms will undoubtedly provide a simple and effective narration of the algorithms described, including KNN, multilinear rgression,, and random forests. We will compare the same dataset using all of the affored mentioned algorithms and project the outcomes for the best performance

CHAPTER-8

TESTING

User Acceptance Testing:

Purpose of Document: The purpose of this document is to briefly explain the test coverage and open issues of the Car resale value prediction project at the time of the release to User Acceptance Testing.

Defect Analysis: This report shows the number of resolved or closed bugs attach security level,and how they were resolved.

	Total Story points	**Duration on Days**	**Sprint Start Date**	**Sprint End Date(Planned)**	**Story pont completed as an planned enddate)**	**Sprintrelease Date(Actual)**
Sprint-1	20	5 days	24 oct 2022	29 oct 2022	20	29 oct 2022
Sprint-2	20	5 days	31 oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	5 days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	5 days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

CHAPTER-9

RESULTS

9.1. PERFORMANCE METRICS:

Random Forest:

The machine learning method in a random forest predicts a value or category by merging the results of many decision trees. The random forest method is a bagging technique variation that use both bagging and feature randomization to generate an uncorrelated forest of decision trees. Performance matrices of Random forest algorithm.

Performance matrices of Random forest regression algorithm:

Accuracy score:

r2_score(Y_test,y_pred)

0.8337646405709781

K-Nearest Neighbors algorithm:

The k-nearest neighbours algorithm, often known as KNN or k-NN, is a non-parametric, supervised learning classifier that employs proximity to classify or predict the grouping of an individual data point.

Performance matrices of KNN algorithm:

Confusion Matrix:

```
array([12, 0, 0 , …..,  0,  0 , 0],
       [0,   0,  0 ….., 0,  0,  0],
       [0,   0,  0 ….., 0,  0,  0],
        ……..
       [0,   0,  0 ….., 0,  0,  0],
       [0,   0,  0 ….., 0,  0,  0],
       [0,   0,   0 …..,0,  0,  0], dtype=int64)
```

Multi-linear Regression :

Multiple linear regression is a type of predictive analysis. This type of analysis allows to understand the relationship between a continuous dependent variable and two or more independent variables.The independent variables can be either continuous or categorical and the dependent variable is categorical.

Performance of multi-linear regression:
Accuracy score:

r2_score(pred,Y_test)

0.21758279301950234

Evaluating Performance Of The Models:

When compared all the other algorithms Random Forest Algorithm has the highest accuracy of 0.8337646405709781using this algorithm ,we obtain the prediction for the car resale value price

Accuracy score:
r2-Score:

0.8337646405709781

CHAPTER-10

ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- The biggest benefit of purchasing a used car is the price, which is frequently much lower discover more about what to anticipate from a dealer or private sale as well as the typical current resale value of the model consumers typically have more alternatives and better funding sources if you acquire a certified pre-owned car from a dealership. Moreover, you can bring your vehicle to the dealership's automotive repair facility for maintenance or repairs.
- One can charge less if you purchase from an owner because you will deal with them directly and avoid paying any fees that might be linked with a dealership purchase while using the "k nearest neighbour, a multilinear regression technique, random forest The biggest victim of depreciation and predit proper exact value is someone else. Buy a used car that has been certified.
- The used automobile has been repaired dealerships provide consumers models that are under five years old that have undergone rigorous inspections, an extended warranty, and other incentives.

DISADVANTAGES:

- Purchasing a secondhand car has several drawbacks despite the low cost. In particular, until you purchase a report, you are unaware of the vehicle past you might not learn everything there is to know about the car even though.
- detectors or replacement components you might not be aware of the repairs that the prior owner did or how they were done. Aftermarket components may restrict operation or provide you with issues in the future.
- Refurbished automobiles might not include the latest newest advancements, reliability, or safety improvements.
- Without certainty Visitors won't get a protection to cover car damage unless you buy a pre-owned automobile from a dealer that is Communications plan Increased repair costs could come from it now.

CHAPTER-11

CONCLUSION

Data preparation and missing value handling are done first, then exploratory analysis, model development, and model assessment are done. This project can help in forecasting the accuracy value price on used based on used car qualities when we get a better accuracy score and other performance indicators on used car value test-set.

CHAPTER-12

FUTURE SCOPE

Future integration of this machine learning model with different websites that can supply real-time data for price prediction is possible. We may also include extensive history data on car prices. It may aid in enhancing the machine learning model's accuracy. An Android app can be created as the user interface for communicating with users. We intend to carefully craft deep learning network topologies, employ adaptive learning rates, and train on data clusters rather than the entire dataset for better performance.

CHAPTER-13

APPENDIX

SOURCE CODE:

Home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>y_predict</title>
    <style>
        body{
         background-image:url('./car.jpg');
         background-repeat:no-repeat;
         background-size: cover;
        }
        h1{
            text-align: center;
            color: aliceblue;
        }
        .form{
            text-align: center;
            color: aliceblue;
            margin-top: 200px;
```

```
        }
    </style>
</head>
<body>
    <h1>Car Resale Price prediction </h1>
<form action='form.html' class="form">
<h2>To predict over the used cars price</h2>
<a href="predicts.html">
    <button type="button" class="btn btn-secondary"> Predict</button>
</a>
</form>
</body>
</html>
```

Predict.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>y_predict</title>
</head>
<style>
  .card-registration .select-input.form-control[readonly]:not([disabled]) {
```

```
    font-size: 1rem;
    line-height: 2.15;
    padding-left: .75em;
    padding-right: .75em;
    }
    .card-registration .select-arrow {
    top: 13px;
    }
</style>
<body>
  <section class="h-100 bg-dark">
    <div class="container py-5 h-100">
      <div class="row d-flex justify-content-center align-items-center h-100">
        <div class="col">
          <div class="card card-registration my-4">
            <div class="col-xl-6">
              <div class="card-body p-md-5 text-black">
                <h3 class="mb-5 text-uppercase">Get the Prediction Resale Value of Your Car</h3>
                <div class="row">
                  <div class="col-md-6 mb-4">
                    <div class="form-outline">
                      <input type="number" name="Registration year" class="form-control form-control-lg" />
                      <label class="form-label" for="Registration year">Registration year </label>
                    </div>
```

```
            </div>
            <div class="col-md-6 mb-4">
              <div class="form-outline">
                <input type="number" name="Registration month" class="form-control form-control-lg" />
                <label class="form-label" for="Registration month">Registration month</label>
              </div>
</div>
          </div>
          <div class="row">
            <div class="col-md-6 mb-4">
              <div class="form-outline">
                <input type="number" name="Power of car" class="form-control form-control-lg" />
                <label class="form-label" for="Power of car">Power of car in PS</label>
              </div>
            </div>
            <div class="col-md-6 mb-4">
              <div cls="form-outline">
                <input type="number" id="Kilometers driven" class="form-control form-control-lg" />
                <label class="form-label" for="Kilometers driven">Kilometers driven</label>
              </div>
            </div>
          </div>
```

```
<div class="row">
  <div class="col-md-6 mb-4">
  <select class="Model Type">
    <option value="1">Model Type Name</option>
    <option value="2"></option>
    <option value="3">Golf</option>
    <option value="4">Grand</option>
    <option value="5">Fabia</option>
    <option value="6">3er</option>
    <option value="7">2 Reihe</option>
    <option value="8">Andere</option>
    <option value="9">C Max</option>
    <option value="10">3 Reihe</option>
    <option value="11">Passat</option>
    <option value="12">Navara</option>
    <option value="13">Ka</option>
    <option value="14">Polo</option>
    <option value="15">Twingo</option>
    <option value="16">A klasse</option>
    <option value="17">Scirocco</option>
    <option value="18">5er</option>
    <option value="19">Meriva</option>
    <option value="20">Arosa</option>
    <option value="21">C4</option>
  </select>
</div>
<div class="col-md-6 mb-4">
```

```
  <select class="Brand">
    <option value="1">Brand</option>
    <option value="2">Ford</option>
    <option value="3">BMW</option>
    <option value="4">Honda</option>
    <option value="2">Toyota</option>
    <option value="3">Tesla</option>
    <option value="4">Audi</option>
    <option value="2">Jeep</option>
    <option value="3">Hyundai</option>
    <option value="4">Kia</option>
    <option value="2">Ferrai</option>
    <option value="3">Porsche</option>
    <option value="4">Bently</option>
    <option value="3">Maserati</option>
    <option value="4">Dodge</option>
  </select>
</div>
</div>
<div class="row">
  <div class="col-md-6 mb-4">
  <select class="Fuel Type ">
    <option value="1">Fuel Type</option>
    <option value="2">Petrol</option>
    <option value="3">Diesel</option>
    <option value="4">Not Declared</option>
    <option value="5">LPG</option>
```

```
      <option value="6">CNG</option>
      <option value="7">Hybrid</option>
      <option value="8">Electric</option>
    </select>
  </div>
  <div class="col-md-6 mb-4">
    <select class="Vehicle type">
      <option value="1">Vehicle type</option>
      <option value="2">Volkswagen</option>
      <option value="3">SUV</option>
      <option value="4">Cabrio</option>
      <option value="5">Kombi</option>
      <option value="6">Andere</option>
      <option value="7">Coupe</option>
      <option value="8">Crossover</option>
      <option value="9">Sports Car</option>
      <option value="10">Compact Car</option>
      <option value="11">Pony Car</option>
      <option value="12">Sedan</option>
      <option value="13">Hatchback</option>
    </select>
  </div>
</div>
    <div class="form-check form-check-inline mb-0 me-4">
      <input class="form-check-input" type="radio" name="Gear type"
        value="option1" /><br>
      <label class="form-check-label" for="GeartypeManual">Manual</label>
```

```
            </div>

            <div class="form-check form-check-inline mb-0 me-4">
              <input class="form-check-input" type="radio" name="Gear type"
                value="option2" />
              <label                                    class="form-check-label"
for="GeartypeAutomatic">Automatic</label>
            </div>

            <div class="form-check form-check-inline mb-0">
              <input class="form-check-input" type="radio" name="Gear type"
                value="option3" />
              <label          class="form-check-label"          for="Geartypeother">Not-
declared</label>
            </div>

          </div>


          <div class="d-md-flex justify-content-start align-items-center mb-4 py-2">

            <h6 class="mb-0 me-4">Your car is repaired or damaged</h6><br>

            <div class="form-check form-check-inline mb-0 me-4">
              <input class="form-check-input" type="radio" name="Your car is repaired
or damaged"
                value="option1" />
```

```
                <label class="form-check-label" for="Repaired">Yes</label>
              </div>

              <div class="form-check form-check-inline mb-0 me-4">
                <input class="form-check-input" type="radio" name="Your car is repaired or damaged"
                  value="option2" />
                <label class="form-check-label" for="Notrepaired">No</label>
              </div>



            </div>

            <div class="form-outline mb-4">

            <div class="d-flex justify-content-end pt-3">
              <button type="button" class="btn btn-light btn-lg">Reset all</button>
              <a href="./value.html">
              <button type="button" class="btn btn-warning btn-lg ms-2">Predict Values</button>
            </a>
              </div>

            </div>
          </div>
        </div>
      </div>
```

```
        </div>
      </div>
    </div>
  </section>
</html>
```

Value.html

```
DOCTYPE html>
<html lang="en">
    <head>
        <title>Car Price Prediction</title>
        <meta charset="utf-8">
        <meta name="viewport" content="width=device-width, initial-scale=1">
        <link rel="stylesheet" href="/static/main.css">
      </head>

<body>

    <div class="container1">
        <h2></h2>
        <div class="alert alert-info" role="alert">
          <strong>Pridicted price of your car is : {{output}}</strong>
        </div>

</body>
</html>
```

App.py

```
from flask import Flask, request, render_template
import pickle

app = Flask(_name_)  # initialising flask app
m=open('C:\\Users\\ishu\\Predicting-the-Price\\model', 'rb')
model = pickle.load(m) # load ml model

@app.route('/', methods=['GET'])
def home():
    return render_template('index.html')



@app.route('/', methods=['POST', 'GET'])
def predict():
    if request.method == 'POST':
        present_price = float(request.form['price'])
        car_age = int(request.form['age'])
        seller_type = request.form['seller']
        fuel_type = request.form['fuel']
        transmission_type = request.form['transmission']

        if fuel_type == 'Diesel':
            fuel_type = 1
        else:
```

```
        fuel_type = 0

    if seller_type == 'Individual':
        seller_type = 1
    else:
        seller_type = 0

    if transmission_type == 'Manual':
        transmission_type = 1
    else:
        transmission_type = 0

    #model = pickle.load(open('model', 'rb'))  # load ml model
    prediction = model.predict([[present_price, car_age, fuel_type, seller_type, transmission_type]])
    output = round(prediction[0]-25, 1)

    return render_template('value.html', output="{} Lakh".format(output))
if _name_ == '_main_':
    app.run(debug=True)
```

Car resale value final.py

Importing Libraries

```
import pandas aspd
import num py as np
```

```
import matplot lib as plt
from sklearn.preprocessing import LabelEncoder
import pickle
import warnings
warnings.filterwarnings('ignore')
from sklearn.tree import DecisionTree Classifier
from sklearn.ensemble import GradientBoosting Classifier, Random Forest Classifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.model_selection import Randomized Search CV
Read the Dataset
df = pd.read_csv("Data/autos.csv",header=0, sep=',',encoding='Latin1')
df.head()
```

Visualising the Data using EDA

```
Import seaborn as sns
Import matplotlib.pyplot as plt
%matplotlib inline
```

Uni-Variate Analysis

```
f =plt.figsize=(8,8)
sns.countplot(df.notRepairedDamage)
f =plt.figsize=(8,5)
sns.countplot(df.kilometer)
f =plt.figsize=(18,8)
sns.countplot(x='vehicleType',data=df)
f =plt.figsize=(8,5)
sns.countplot(x='fuelType',data=df)
sns.countplot(x='brand',data=df)
```

```
plt.scatter(df.index,df['yearOfRegistration'])
plt.show()

plt.hist(df['monthOfRegistration'])
 (array([62236., 22403., 36170., 30918., 30631., 62125., 23765., 25074.,
27337., 50869.]),
 array([ 0. ,  1.2,  2.4,  3.6,  4.8,  6. ,  7.2,  8.4,  9.6, 10.8, 12. ]),
 )

df['powerPS'].plot(kind='density')
```

Bivariate Analysis

```
sns.countplot(df['brand'],hue=df['notRepairedDamage'])
sns.countplot(df['model'],hue=df['vehicleType'])]

plt.scatter(df.kilometer,df.price)
plt.figure(figsize=(18,5))
plt.subplot(1,4,1)
sns.countplot(df['vehicleType'])
plt.subplot(1,4,2)
sns.countplot(df['kilometer'])
plt.show()
```

Multivariate Analysis

```
plt.figure(figsize= (5,5))
sns.heatmap(df.corr(),annot=True)
plt.plot(df.yearOfRegistration,df.kilometer,df.price)
sns.boxplot(x = 'brand', y = 'kilometer', data =df)
```

df.plot.line() [22]:

df.hist()

Descriptive Analysis

df.describe()

Cleaning the Dataset

print(df.seller.value_counts())

privat 371525

gewerblich 3

Name: seller, dtype: int64

df[df.seller!= 'gewerblich']

df=df.drop('seller',1)

print(df.offerType.value_counts())

Angebot 371516

Gesuch 12

Name: offerType, dtype: int64

df[df.offerType!= 'Gesuch']

df.drop('offerType',1)

print(df.shape)

(371528, 19)

df=df[(df.powerPS>50)&(df.powerPS<900)]

print(df.shape)

(319709, 19)

df=df[(df.yearOfRegistration>=1950)&(df.yearOfRegistration<2017)]

print(df.shape)

(309171, 19)

```
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','powerPS','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'),('small car','convertible','combination','others'),inplace=True)
new_df.notRepairedDamage.replace(('ja','nein'),('Yes','No'),inplace=True)
```

Checking For Null Values

```
df.isnull().sum()
offerType               0
price                   0
vehicleType         11422
yearOfRegistration      0
gearbox              5298
powerPS                 0
model               11799
kilometer               0
monthOfRegistration     0
fuelType            15887
brand                   0
```

notRepairedDamage 43481

dtype: int64

```
sns.heatmap(df.isnull(),yticklabels=False,cbar=True,cmap='Accent')
```

Removing the outliers

```
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['model'].fillna(value='not-declared',inplace=True)
```

Saving the cleaned dataset

```
new_df.to_csv("autos_prepocessed.csv")
```

Label encoding the categorical data

```
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)
gearbox ;LabelEncoder()
notRepairedDamage ;LabelEncoder()
model ;LabelEncoder()
brand ;LabelEncoder()
fuelType ;LabelEncoder()
vehicleType ;LabelEncoder()
Index(['price', 'yearOfRegistration', 'powerPS', 'kilometer',
      'monthOfRegistration', 'gearbox_labels', 'notRepairedDamage_labels',
      'model_labels', 'brand_labels', 'fuelType_labels',
      'vehicleType_labels'],
dtype='object')
```

Splitting Data into IndependantAnd Dependant Variables

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

Splitting Data Into Train And Test

```
fromsklearn.model_selectionimportcross_val_score,train_test_split
X_train,X_test,Y_train,Y_test=train_test_split(X,Y,test_size=0.3,random_state)
```

Random forest regression model

```
fromsklearn.neighborsimportKNeighborsClassifier
fromsklearn.metricsimportaccuracy_score
knn=KNeighborsClassifier()
knn.fit(X_train,Y_train)
KNeighborsClassifier()
```

Choose The Appropriate Model

```
#predicting the values of test
pred_test=knn.predict(X_test)
pred_test
array([4399,  900, 2400, ..., 1250, 2799,  700], dtype=int64)
#the accuracy for test set
accuracy_score(pred_test,Y_test)
0.018331059899011654
```

Confusion Matrix

```
from sklearn.metrics import confusion_matrix
confusion_matrix(pred_test,Y_test)
array([[12,  0,  0, ...,  0,  0,  0],
       [ 0,  0,  0, ...,  0,  0,  0],
       [ 0,  0,  0, ...,  0,  0,  0],
       ...,
       [ 0,  0,  0, ...,  0,  0,  0],
       [ 0,  0,  0, ...,  0,  0,  0],
[ 0,  0,  0, ...,  0,  0,  0]], dtype=int64)
plt.figure(figsize=(5,4))
x_as=range(len(Y_test))
```

```
plt.scatter(pred_test,Y_test,label="Original")
plt.scatter(x_as,pred_test,label="Predicted")
plt.title("Actual car resale value prediction")
plt.legend()
plt.show()
```

Save The Model

```
filename= 'Data/resale_model.sav'
pickle.dump(knn, open(filename, 'wb'))
```

Random Forest Regression

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

Choose The Appropriate Model

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

Check The Metrics Of The Model

```
y_pred=regressor.predict(X_test)


print(r2_score(Y_test,y_pred))
0.8337646405709781
```

Save The Model

```
filename= 'Data/resale_model.sav'
pickle.dump(regressor, open(filename, 'wb'))
```

Multi-Linear Regression model

From sklearn.linear_model impor tLinearRegression

mlr=LinearRegression()

Choose The Appropriate Model

LinearRegression()

pred=mlr.predict(X_test)

pred

array([[7220.37572008],

[-970.58855267],

[1262.52734917],

...,

[1383.22556109],

[2628.34656608],

[-688.76113019]])

pred.astype(int)

array([[7220], [-970], [1262] [1383],[2628], [-688]])Y_testarray([[10000],[1500],[2699],[1300],[5000],[777]], dtype=int64)

Model Evaluation

from sklearn.metrics import r2_score

r2_score(pred,Y_test)

0.21758279301950234

plt.figure(figsize=(7,7))

x_as=range(len(Y_test))

plt.plot(x_as,Y_test,label="Original")

plt.plot(x_as,pred,label="Predicted")

plt.title("Actual car resale value prediction")

```
plt.legend()

plt.show()
```

Save The Model

```
filename= 'Data/resale_model.sav'
pickle.dump(mlr , open(filename, 'wb'))
```

Github Link: https://github.com/IBM-EPBL/IBM-Project-22893-1659860293