CAR RESALE VALUE PREDICTION

A PROJECT REPORT

TEAM ID:

PNT2022TMID15627

TEAM MEMBERS:

1. ARAVINDH R (927619BEC4010)
2. ARSHATH AHAMED N (927619BEC4011)
3. ARUN A (927619BEC4012)
4. DEEPAK KUMAR S (927619BEC4025)

INDUSTRY MENTORS NAME:

1. Prof Swetha

FACULTY MENTOR NAME:

Mr. ARUNPRATHAP S

INDEX

Project Report

1. INTRODUCTION

1.1 Project Overview

In this project we have used different algorithms with different techniques for developing Car resale value prediction systems considering different features of the car. In a nutshell, car resale value prediction helps the user to predict the resale value of the car depending upon various features like kilometres driven, fuel type, etc.

This resale value prediction system is made for general purpose to just predict the amount that can be roughly acquired by the user.

1.2 Purpose

Used car resale market in India was marked at 24.2 billion US dollars in 2019. Due to the huge requirement of used cars and lack of experts who can determine the correct valuation, there is an utmost need of bridging this gap between sellers and buyers. This project focuses on building a system that can accurately predict a resale value of the car based on minimal features like kms driven, year of purchase etc. without manual or human interference and hence it remains unbiased.

The main idea of making a car resale value prediction system is to get hands-on practice for python using Data Science. Car resale value prediction is the system to predict the amount of resale value based on the parameters provided by the user. User enters the details of the car into the form given and accordingly the car resale value is predicted.

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.

2. LITERATURE SURVEY

2.1 Existing problem

The rise of online websites and other tools like it have made it easier for both buyers and sellers to get a better understanding of the factors that determine the market value of a used car. Based on a set of factors, Machine Learning algorithms may be used to forecast the price of any automobile. The cost is calculated using the amount of characteristics. They used linear regression and lasso regression to develop a price model for used automobiles in comparative research. The main goal of this study is to discover the best predictive model for estimating the price of a used car.

In existing model, it was to estimate the cost of the used cars using the K nearest neighbour algorithm which is simple and suitable for small data set. Here, they have collected a used cars dataset and analysed the same. The data was trained by the model and examined the accuracy of the model among different ratios of trained and test set. The same model is cross-validated for assessing the performance of the model using the K- Fold method which is easy to understand and implement. They have used the K nearest Neighbour algorithm and got accuracy 85% where the accuracy of linear regression is 71%. The proposed model is also validated with 5 and 10 folds by using K Fold Method.

2.2 References

1. CAR PRICE PREDICTION USING MACHINE LEARNING [KETAN AGRAHARI, AYUSH CHAUBEY ET AL, 2021]

2. VEHICLE RESALE PRICE PREDICTION USING MACHINE LEARNING": [B.LAVANYA, SK.RESHMA, N.NIKITHA, M.NAMITHA , L.KANYA KUMARI, S.KISHORE BABU, 2021]

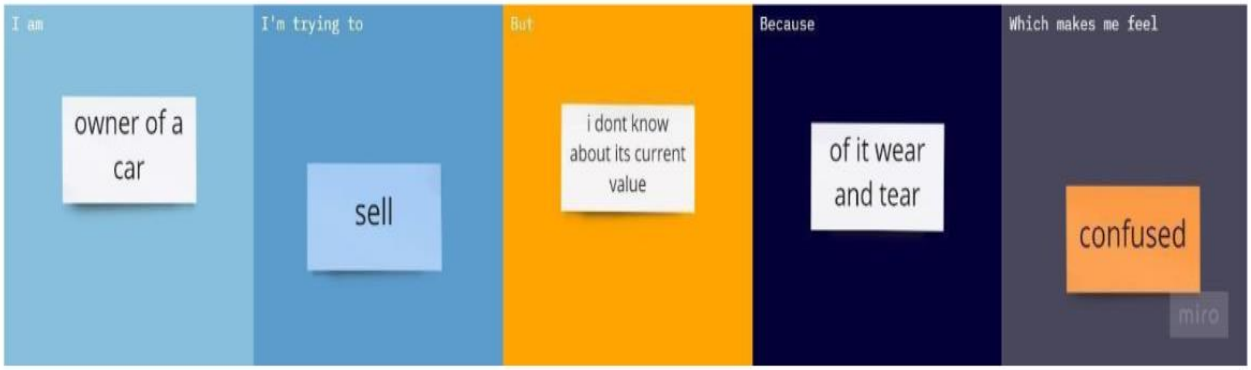
3. AN EXPERT SYSTEM OF PRICE FORECASTING FOR USED VEHICLES USING ADAPTIVE NEURO-FUZZY INFERENCE [WU, ET AL, 2009]

4. USED CAR PRICE PREDICTION USING K-NEAREST NEIGHBOR BASED MODEL [SAMRUDDHI, ASHOK KUMAR, 2020]

2.3 Problem Statement Definition

The research objective of this study is to predict used cars prices, using data mining techniques, by scraping data from websites that sell used cars, and analysing the different aspects and factors that lead to the actual used car price valuation. To enable consumers to know the actual worth of their car or desired car, by simply providing the program with a set of attributes from the desired car to predict the car price. The purpose of this study is to understand and evaluate used car prices and to develop a strategy that utilizes data mining techniques to predict used car prices.

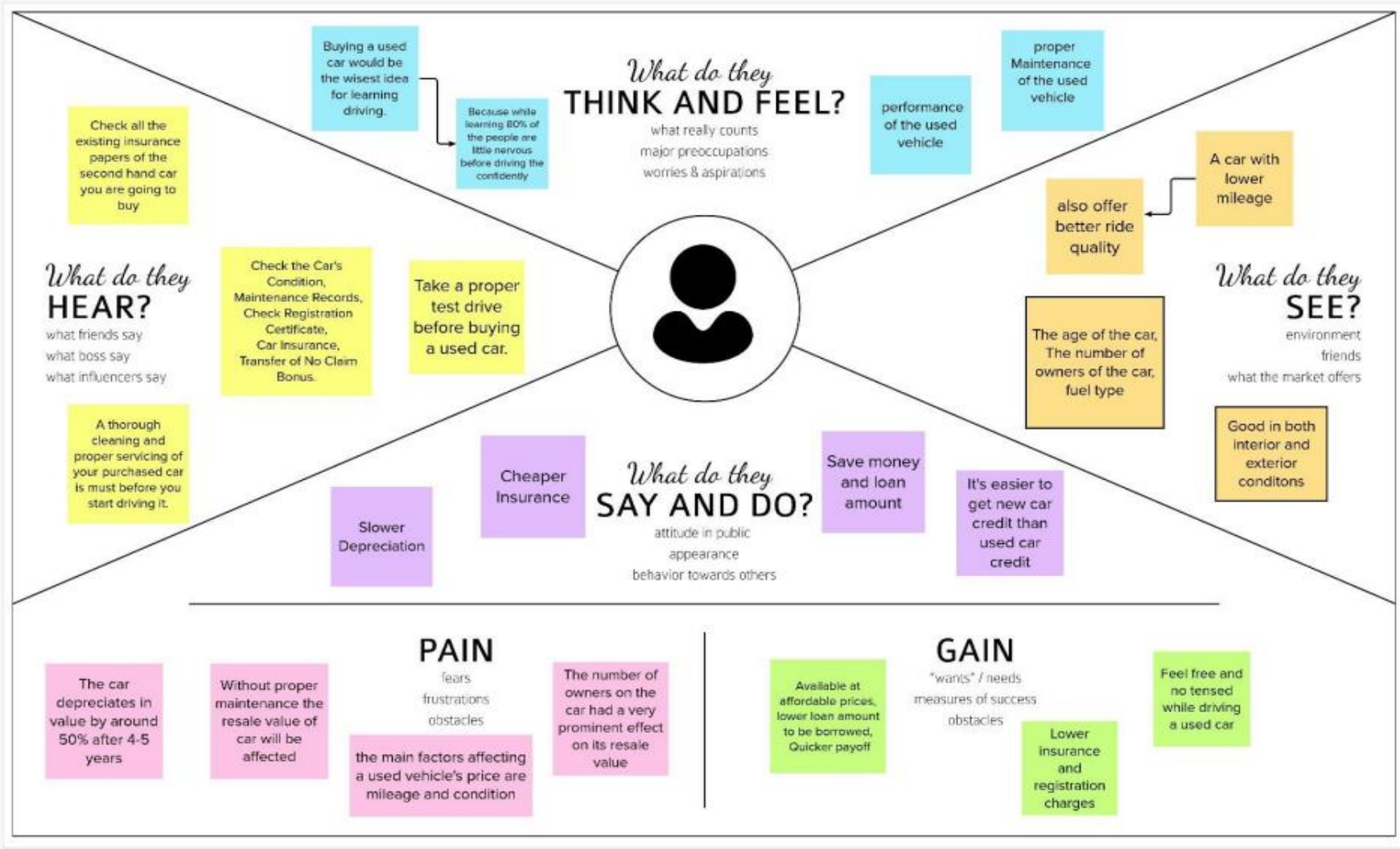
PROBLEM STATEMENT:



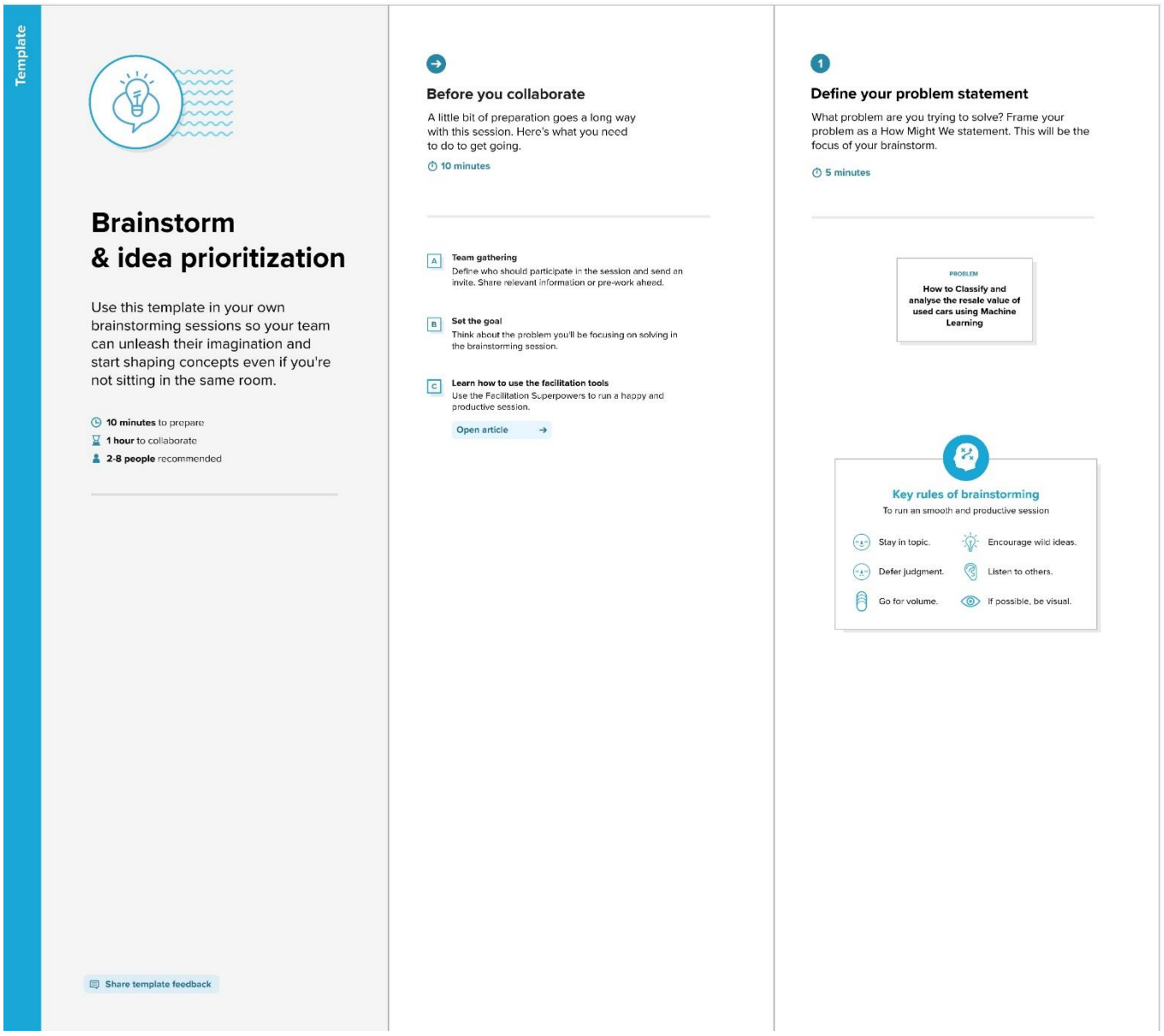
Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Owner of a car	Sell	I don't know about its current value	Of it wear and tear	Confused
PS-2	Owner of a car	Find its resale value	I am concerned about the minor dents	Of a small accident	A little worried

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch] icon to start drawing!

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

Grouping based on the dataset

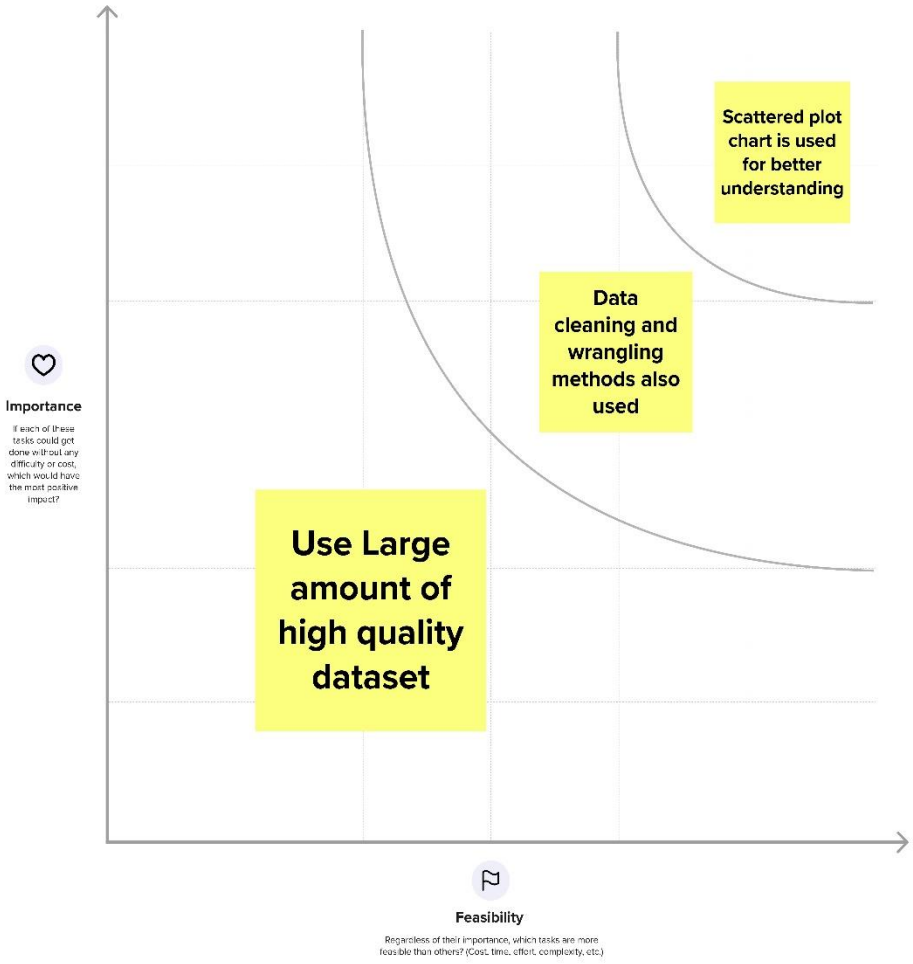
Grouping based on model

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 as feasible.

20 minutes



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

A **Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B **Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

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	Description
1.	Problem Statement (Problem to be solved)	The objective of this project is to predict the price of used cars using the various Machine Learning models by using User Interface (UI).
2.	Idea / Solution description	To train the system with the dataset using a regression model and it will be integrated to the web-based application where the user is notified with the status.
3.	Novelty / Uniqueness	By using the optimal regression model to predict the value in a less amount to time and predict its value.
4.	Social Impact / Customer Satisfaction	The customer can get an idea about the resale value of their car and to determine the actual value of the used car by their own within few minutes by using various features such as age of car, brand, fuel type, kilometres driven .
5.	Business Model (Revenue Model)	The web-based application has a friendly UI forthe customer to enter their vehicles detail and the system predicts the value within few seconds.
6.	Scalability of the Solution	Machine learning approaches, this project proposed a scalable framework for predicting values for different type of used cars. The solution given by the trained system is efficientand is nearly accurate value of the vehicle.

3.4 Problem Solution fit

The problem solution Fit simply means that you have identified a customer-related issue and that the resolution you have developed genuinely addresses the issue. It assists business owners, marketers, and corporate innovators in seeing behavioural trends and understanding what would be successful.

Purpose:

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

Project Title: Car Resale Value Prediction | **Project Design Phase-I - Solution Fit Template** | **Team ID:** PNT2022TMID15627

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) — CS

Who is your customer?
i.e. working parents of 0-5 y.o. kids

Both used car sellers and buyers.

6. CUSTOMER CONSTRAINTS — CC

What constraints prevent your customers from taking action or limit their choices of solutions? i.e. spending power, budget, no cash, network connection, available devices.

- To determine the worthiness of the car by their own within few minutes
- A loss function is to be optimized by spending money for dealers,brokers to buy or sell a car.

5. AVAILABLE SOLUTIONS — AS

Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking

- In the past User cannot find the value of used car buy their own without prior knowledge about cars.
- A person who don't know much about the car can also make predictions for used cars easily.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS — J&P

Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.

To build a supervised machine learning model using regression algorithms for forecasting the value of a vehicle based on multiple attributes such as Condition of Engine, Year of Registration, Kilometers, Number of Owner

9. PROBLEM ROOT CAUSE — RC

What is the real reason that this problem exists? What is the back story behind the need to do this job?
i.e. customers have to do it because of the change in regulations.

- The price predicted by the dealers or brokers for used car is not trustful
- Users can predict the correct valuation of the car remotely without human intervention like car dealers.
- User can eliminate the valuation predicted by the dealer.

7. BEHAVIOUR — BE

What does your customer do to address the problem and get the job done?
i.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (i.e. Greenpeace)

The History of Your Car's condition and documents produced by them will be Suspicious. The model is to be built would give the nearest value of the vehicle by eliminating anonymous value predicted by using humans.

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS — TR

What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.

Users can predict the correct valuation of the car by their own like Olxcars, Cars24 and other car resale value prediction websites by using model,year,owner,etc

10. YOUR SOLUTION — SL

If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality.
If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.

- **The main aim of this project is to predict the price of used cars using the Machine Learning(ML) algorithms and collection data's about different cars.**

8. CHANNELS of BEHAVIOUR — CH

8.1 ONLINE
What kind of actions do customers take online? Extract online channels from #7

8.2 OFFLINE
What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.

- **Customer should predict the worth of the car by using different parameters given by the owner.**
- **User Should confirm the details provided about the vehicle in RTO online.**

Identify strong TR & EM

4. EMOTIONS: BEFORE / AFTER — EM

How do customers feel when they face a problem or a job and afterwards?
i.e. lost, insecure > confident, in control - use it in your communication strategy & design

Before:

- **User will be in fear about the biased values predicted by the humans based on the condition of the car.**

After:

- **User can determine the worthiness of the car by their own without human intervention.**

- **The project should take parameters related to used car as inputs and enable the customers to make decisions by their own.**

- **User can decide by seeing the exterior and interior condition of the car.**
- **User can test the performance of the car and to buy it up in a affordable price based on its condition.**

4. REQUIREMENT ANALYSIS

4.1 Functional requirements

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through New use portal Registration through gmail
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	Libraries & Modules	**NumPy** for scientific computing **Pandas** for using open-source data analysis **Scikit-learn** for using machine learning techniques
FR-4	Frameworks	Flask framework for building Web applications
FR-5	IDE	PyCharm IDE Jupyter Notebooks for using Machine learning related techniques.

4.2 Non-Functional requirements

Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Estimating the value at resale
NFR-2	**Security**	Offering the website with security policy
NFR-3	**Reliability**	Because it could estimate the worth of any type of car, it would be highly reliable.
NFR-4	**Performance**	Implementing a website with heavier performance, that does not lag on type pf input data
NFR-5	**Availability**	The website would be easily to anyone, since it is a website its accessible form any part of the world

5. PROJECT DESIGN

5.1 Data Flow Diagrams

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.

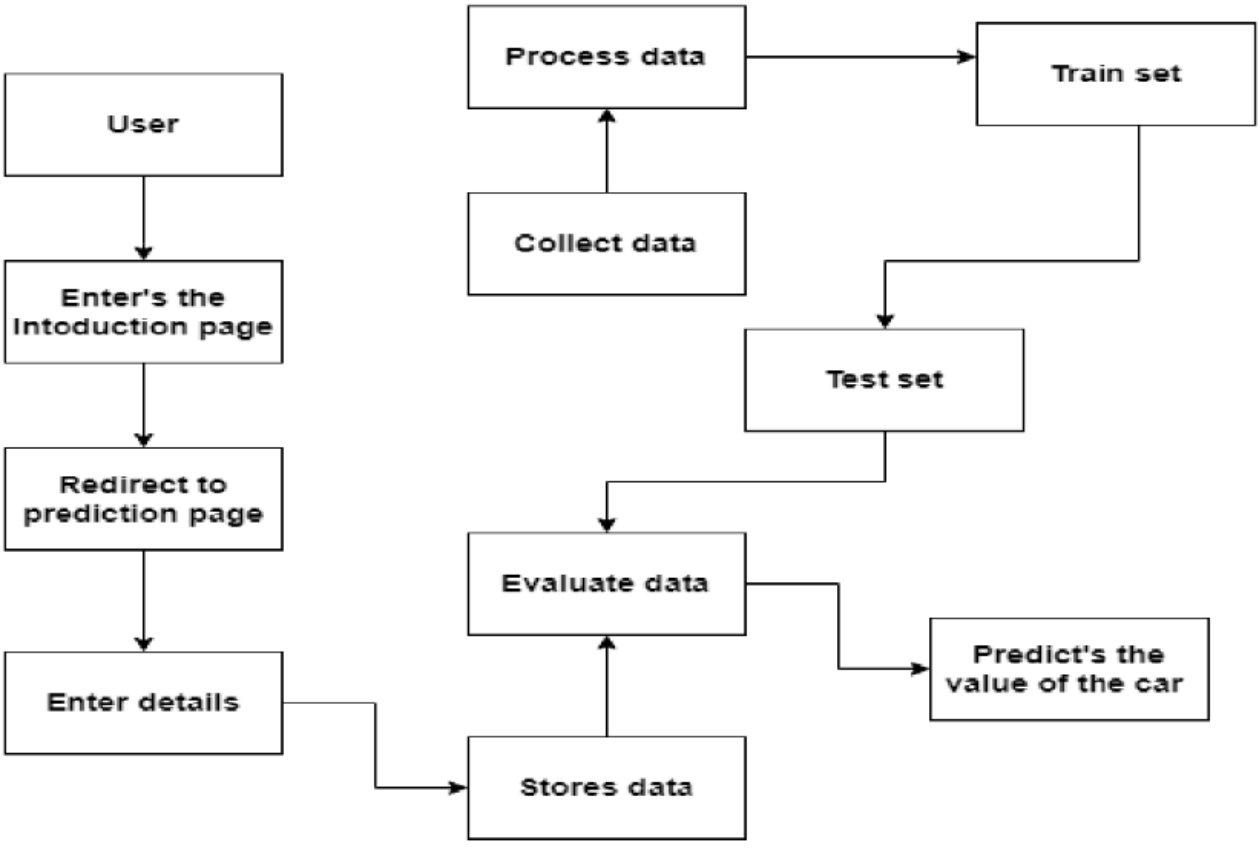
These diagrams are used as visualization tool to help the audience get a better idea of what exactly is going on in the system. The DFDs are use to :

(i) discuss with the user a diagrammatic interpretation of the process in the system and clarify what is currently being performed.

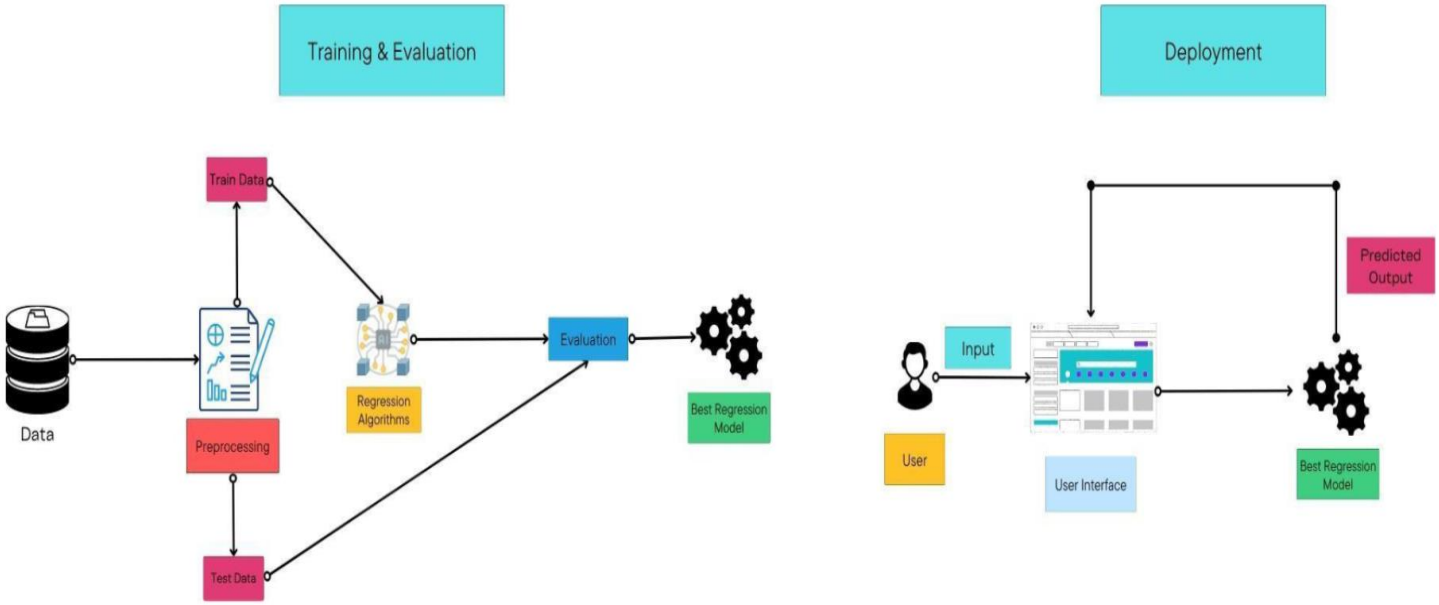
(ii) determine what the new system should be able to do and what information is required for each different process the should be carried out.

(iii) Check that the completed system conforms to its intended design.

(iv) provides easy presentation and communication between technical and non technical staff.



5.2 Solution & Technical Architecture



S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Python
3.	Database	Data Type, Configurations etc.	NoSQL
4.	Cloud Database	Database Service on Cloud	IBM DB2
5.	File Storage	File storage requirements	IBM Block Storage
6.	Machine Learning Model	Purpose of Machine Learning Model	Regression model
7.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		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-1
	Dashboard					
Customer (Web user)	Registration	USN-1	As a user, I can register and enter my car details	I can enter the details of my asset	Medium	Sprint-1

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

TITLE	DESCRIPTION	DATE
LITERATURE SURVEY	Analysis of the use case chosen by referring various journals and research papers	20 SEPTEMBER 2022
INFORMATION GATHERING	Collecting data by consulting technical documents, research articles, etc.	24 SEPTEMBER 2022
PREPARATION OF EMPATHY MAP	Prepare a list of problem statements in publications, a canvas for an empathy map to capture the user's gains and pains, etc.	10 OCTOBER 2022
IDEATION	List the ideas generated during the brainstorming session and rank the top three according to relevance and viability.	30 SEPTEMBER 2022
PROPOSED SOLUTION	Create a proposal for a solution that details its innovation, viability as a business idea, social impact, scalability, and other factors.	12 OCTOBER 2022
PROBLEM SOLUTION FIT	Get a problem-solution-fit document ready.	10 OCTOBER 2022
SOLUTION ARCHITECTURE	Document the solution architecture.	12 OCTOBER 2022
CUSTOMER JOURNEY	Create customer journey maps to comprehend how users engage with and use the application from entry to exit.	15 OCTOBER 2022
FUNCTIONAL REQUIREMENT	Document the functional requirements.	14 OCTOBER 2022
DATA FLOW DIAGRAMS	Create the data flow diagrams, then submit them for evaluation.	13 OCTOBER 2022
TECHNOLOGY ARCHITECTURE	Create the diagram of the technological architecture.	20 OCTOBER 2022
PREPARE MILESTONE & ACTIVITY LIST	Create a list of the project's milestones and activities.	28 OCTOBER 2022
PROJECT DEVELOPMENT - DELIVERY OF SPRINT-1, 2, 3 & 4	Create the code, develop it and submit it after testing it.	10 NOVEMBER 2022

6.2 Sprint Delivery Schedule

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Pre-processing of data	USN-1	Collect Dataset	5	High	Arun A
Sprint-1		USN-2	Import required libraries , Read & clean the data sets.	5	High	Aravindh R
Sprint-2	Building the Model	USN-1	Split data into independent and dependent variables	4	High	Deepak Kumar S
Sprint-2		USN-2	Apply using regression model	2	Medium	Arshath AhamedN
Sprint-3	Application building	USN-1	Build python flask application and HTML page	5	High	Aravindh R
Sprint-3		USN-2	Execute and test the application	1	Low	Deepak Kumar S
Sprint-4	Training the model	USN-1	Train machine learning model that was built	5	High	Arun A
Sprint-4		USN-2	Integrate flask	5	High	Arshath AhamedN

7. CODING & SOLUTIONING

(Explain the features added in the project along with code)

```
from flask import  Flask,render_template,request
import pickle
import numpy as np

app = Flask(__name__)
model=pickle.load(open("resale_value_pickle_file.sav","rb"))
@app.route("/")
def home():
    return render_template("cars-index.html")

@app.route('/submit',methods=["POST","GET"])
def prediction():
    if request.method=="POST":

        yearofRegistration=request.form["yearofRegistration"]

        monthofRegistration=request.form["monthofRegistration"]

        powerPS = request.form["powerPS"]

        kilometer = request.form["kilometer"]

        notRepairedDamage = request.form["notRepairedDamage"]

        if notRepairedDamage == "Yes":
            notRepairedDamage =1

        elif notRepairedDamage == "No":
            notRepairedDamage =0

        elif notRepairedDamage == "not-declared":
            notRepairedDamage =2

        brand = request.form["brand"]

        if brand == "audi":
            brand = 1

        elif brand == "jeep":
            brand = 14

        elif brand == "volkswagen":
            brand = 38

        elif brand == "skoda":
            brand = 31
```

```
elif brand == "bmw":
    brand = 2

elif brand == "nissan":
    brand = 23

elif brand == "renault":
    brand = 27

elif brand == "ford":
    brand = 10

elif brand == "honda":
    brand = 11

elif brand == "mercedes_benz":
    brand = 20

elif brand == "toyota":
    brand = 36

elif brand == "hyundai":
    brand = 12

elif brand == "kia":
    brand = 15

elif brand == "peugeot":
    brand = 25

elif brand == "mitsubishi":
    brand = 22

elif brand == "fiat":
    brand = 9

elif brand == "volvo":
    brand = 39

elif brand == "suzuki":
    brand = 35

elif brand == "porsche":
    brand = 26

elif brand == "dacia":
    brand = 6

gearbox_feat=request.form["gearbox_feat"]
if gearbox_feat == "manual":
    gearbox_feat =1
```

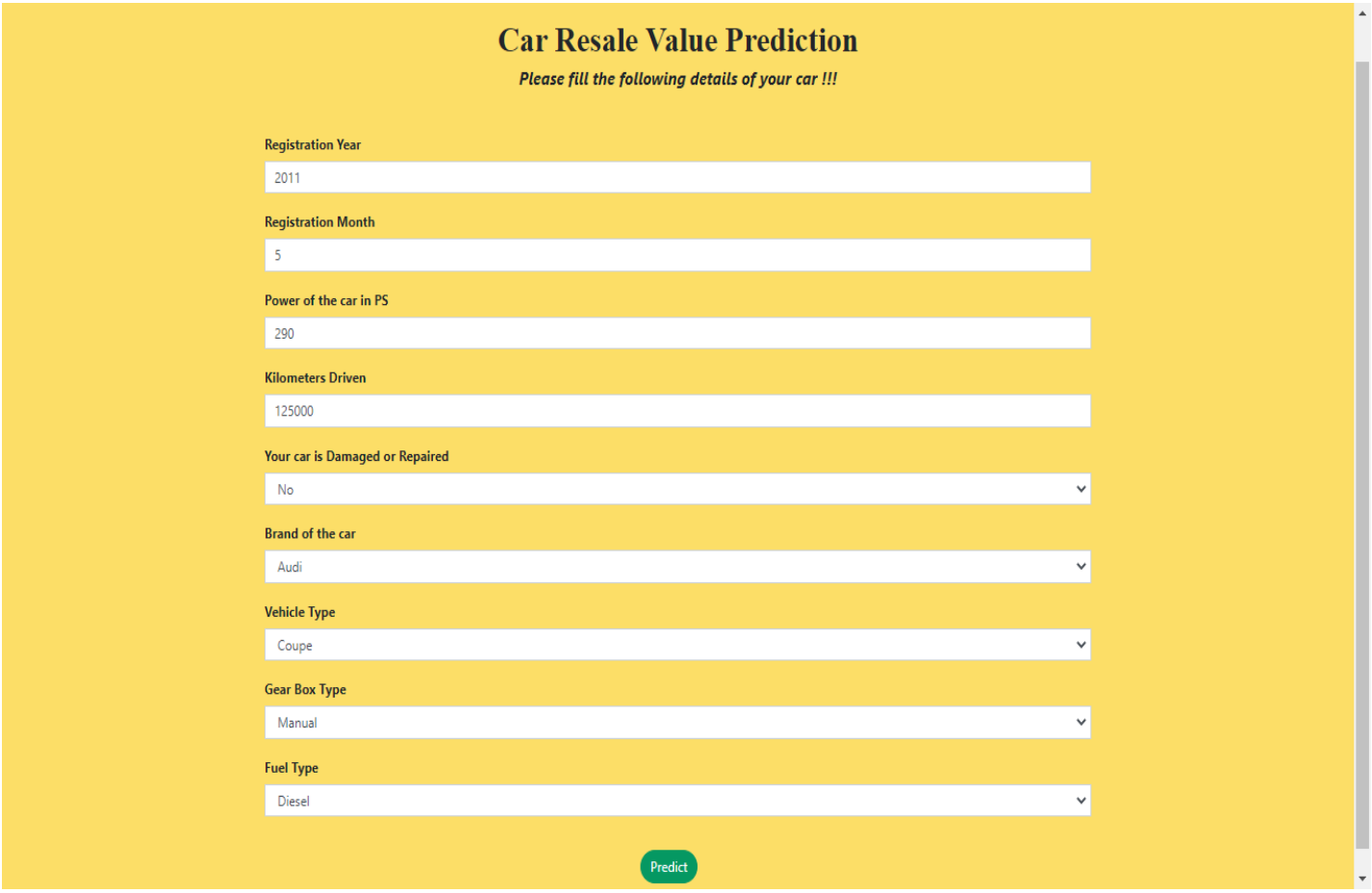
```
    elif gearbox_feat == "automatic":
        gearbox_feat =0

    fuelType_feat = request.form["fuelType_feat"]
    if fuelType_feat=="petrol":
        fuelType_feat=1

    elif fuelType_feat=="diesel":
        fuelType_feat=3

    elif fuelType_feat=="lpg":
        fuelType_feat=4

    elif fuelType_feat=="hybrid":
        fuelType_feat=6

    elif fuelType_feat=="cng":
        fuelType_feat=7

    vehicleType = request.form["vehicleType"]
    if vehicleType == "coupe":
        vehicleType = 3

    elif vehicleType == "suv":
        vehicleType = 8

    elif vehicleType == "small car":
        vehicleType = 7

    elif vehicleType == "limousine":
        vehicleType = 4

    elif vehicleType == "bus":
        vehicleType = 0

    elif vehicleType == "combination":
        vehicleType = 1

    elif vehicleType == "others":
        vehicleType = 6

    elif vehicleType == "convertible":
        vehicleType = 7


    int_features = [yearofRegistration,
powerPS,kilometer,monthofRegistration,notRepairedDamage,brand,gearbox_feat,fuelType_feat,vehicleTyp
e]
    features = [np.array(int_features, dtype=int)]
    prediction=model.predict(features)
    return render_template("cars-submit.html",prediction=round(prediction[0],2))
```

```
if __name__=="__main__":
    app.run(debug=True)
```

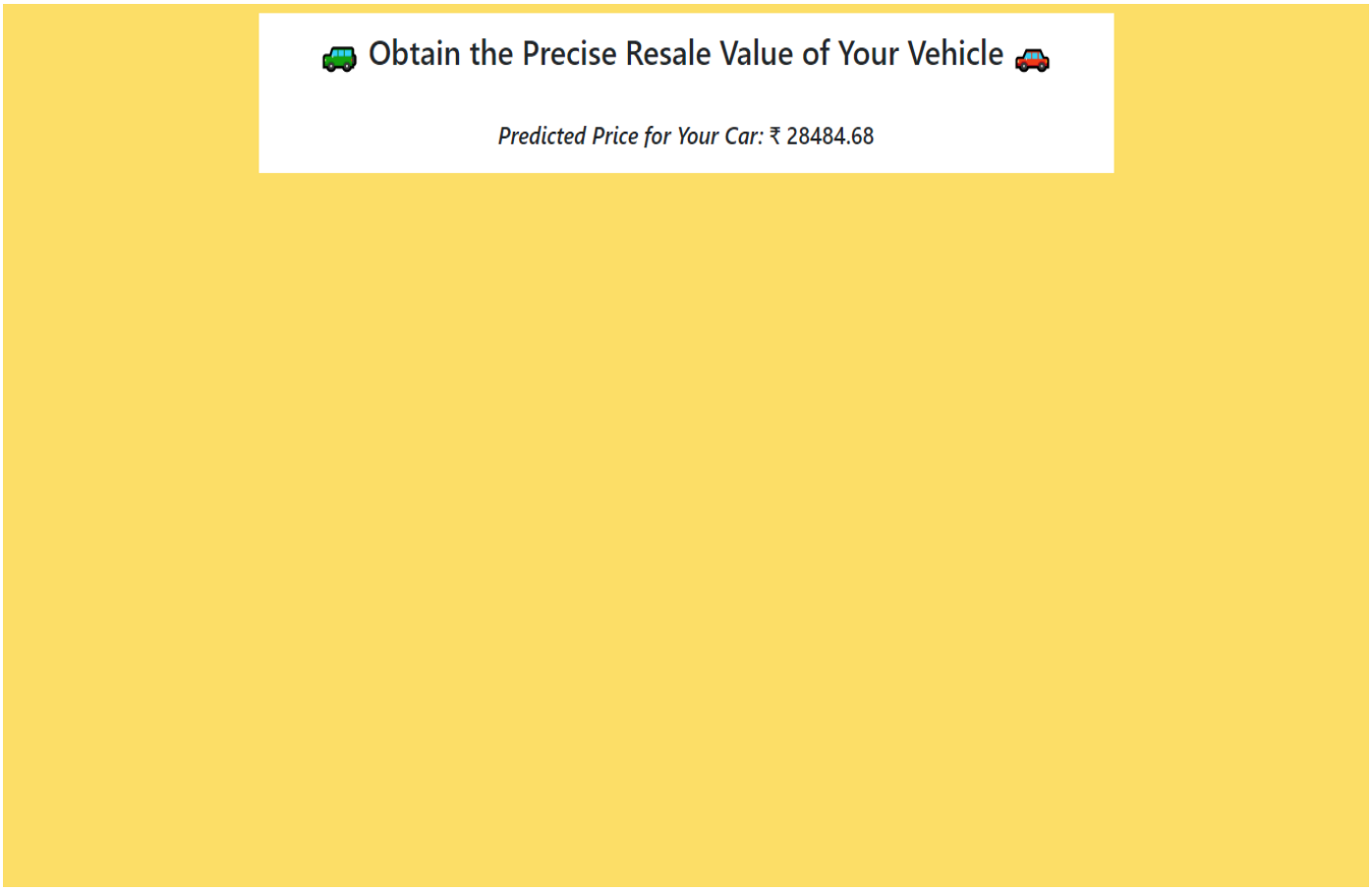
8. TESTING

8.1 Test Cases

1. User Interface (Input Page)



2. OUTPUT PAGE



8.2 User Acceptance Testing

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.

The main **Purpose of UAT** is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

UAT is performed by –

- Client
- End users

On Performing User Acceptance Testing the trained model gives the expected result which in turn meets the business requirements and satisfies all the conditions

UAT is the last chance to identify and rectify defects. Businesses may suffer losses if UAT is not performed properly (or at all). The losses that may occur (by fixing system issues after production), are much more expensive than fixing before production.

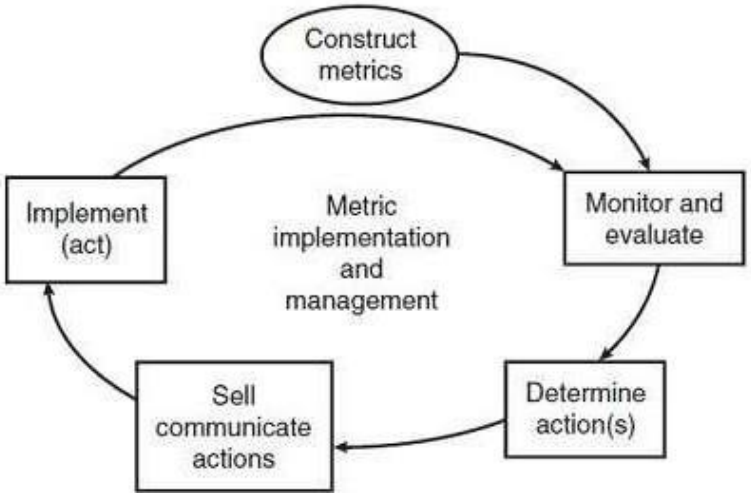
The organization may also lose some reputation as a result of moving defective software to production. Therefore, an UAT is vital. Software testing is a broad field in computing.

9. RESULTS

9.1 Performance Metrics

The following performance measurement necessities are the same whether you're measuring business, service, process, or laboratory variables. Together, they constitute a measurement plan.

- **Definition of purpose:** Why is a measurement being made? What process or variable is being measured? For what will the resulting data be used?
- **Statement of the required measurement performance indicators (accuracy, precision, resolution):** These may be determined by organizational policy, adherence to a published standard or an analysis of the requirements based on use, ability to measure, or more.
- **The unit or variable being measured** and a statement as to why measuring that particular variable supports the purpose of the measurement.
- **An operational definition:** A detailed, yet easily understood, description of the measurement process.
- **An analysis plan:** A typical example is a monthly report that makes comparisons to the previous month, year over year, and year to date. The different time frames provide greater context and allow the data to be presented graphically.
    - A control chart is a simple analysis plan template. It provides a graphical context that shows the continuity of changes over time, plus some analysis (control limits) that enables the viewer to differentiate among common causes, special causes, and random variation.



10. ADVANTAGES

1. Easily identifies trends and patterns

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviours and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

11. DISADVANTAGES

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfil their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

12.CONCLUSION

By performing different ML models, we aim to get a better result or less error with max accuracy. Our purpose was to predict the price of the used cars having 25 predictors and 509577 data entries.

Initially, data cleaning is performed to remove the null values and outliers from the dataset then ML models are implemented to predict the price of cars.

Next, with the help of data visualization features were explored deeply. The relation between the features is examined.

From the below table, it can be concluded that XGBoost is the best model for the prediction for used car prices. XGBoost as a regression model gave the best MSLE and RMSLE values.

Result of Models:

Model	MSLE	RMSLE	Accuracy
Linear regression	0.00243399	0.04933557	59.3051%
Ridge regression:	0.00243399	0.04933553	59.3051%
Lasso regression	0.00243400	0.04933566	59.305%
KNN	0.00144004	0.03794796	76.4681%
Random Forest	0.00077811	0.00077811	87.5979%
Bagging Regressor	0.00143192	0.03784080	76.809%
Adaboost Regressor	0.00084475	0.02906475	86.4084%
XGBoost Regressor	0.00065047	0.02550431	89.6623%

13.FUTURE SCOPE

- ✓ Thus, the Random Forest Regressor Regression model trained by us using IBM Watson Studio with the dataset provided by the mentor gives 95% exact resale value of the car.
- ✓ Need to collect more data and develop the dataset and train more in the model for best results and to consider all the models that are present in the real world.
- ✓ In 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.

14.APPENDIX

14.1 Source Code

Cars-index.html (Input Page) :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <title>Car Resale Value Prediction</title>
   <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
   <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">

   <style>
      body{
         background-color: #fcde67;
      }
      .container{
         background-color: #fcde67;
         margin-top: 50px;
         padding: 20px;
         border-radius: 5px;
      }
      .form-group{
         margin-top: 20px;
      }
      .form-control{
         border-radius: 0px;
      }
      .btn{
         border-radius: 80px;
         margin-top: 20px;
         background-color: #059862;
            border-color: #059862;  }
```

```
        .fa{
          font-size: 30px;
          color: #f2f2f2;
        }

    </style>
</head>
<body>
    <div class="container">
        <h1 class="text-center" style = "font-family:Argent CF"><b>Car Resale Value Prediction</b></h1>
            <h5 style = "text-align:center"><i><strong>Please fill the following details of your car !!!
</strong></i></h5><br>
        <form action="/submit" method="POST">

            <div class="form-group">
                <label for="yearofRegistration"><b>Registration Year</b></label>
                <input type="number" class="form-control" id="yearofRegistration" name="yearofRegistration"
placeholder="Enter Year of Registration">
            </div>
            <div class="form-group">
                <label for="monthofRegistration"><b>Registration Month</b></label>
                <input type="number" class="form-control" id="monthofRegistration"
name="monthofRegistration" placeholder="Enter Month of Registration">
            </div>
            <div class="form-group">
                <label for="powerPS"><b>Power of the car in PS</b></label>
                <input type="number" class="form-control" id="powerPS" name="powerPS" placeholder="Enter
Power of the car in PS">
            </div>
            <div class="form-group">
                <label for="kilometer"><b>Kilometers Driven </b></label>
                <input type="number" class="form-control" id="kilometer" name="kilometer"
placeholder="Enter Kilometers Driven">
            </div>
            <div class="form-group">
                <label for="notRepairedDamage"><b>Your car is Damaged or Repaired</b></label>
                            <select class="form-control" id="notRepairedDamage" name="notRepairedDamage">
                    <option value="Yes">Yes</option>
                    <option value="No">No</option>
                    <option value="not-declared">Not-declared</option>
                </select>
            <div class="form-group">
                <label for="brand"><b>Brand of the car</b></label>
                <select class="form-control" id="brand" name="brand">
                                <option value="audi">Audi</option>
                    <option value="bmw">BMW</option>
                    <option value="dacia">Dacia</option>
                    <option value="fiat">Fiat</option>
                    <option value="ford">Ford</option>
                    <option value="honda">Honda</option>
                    <option value="hyundai">Hyundai</option>
                    <option value="jeep">Jeep</option>
```

```
                <option value="kia">Kia</option>
                <option value="mercedes_benz">Mercedes Benz</option>
                <option value="mitsubishi">Mitsubishi</option>
                <option value="nissan">Nissan</option>
                <option value="peugeot">Peugeot</option>
                <option value="porsche">Porsche</option>
                    <option value="renault">Renault</option>
                <option value="skoda">Skoda</option>
                <option value="suzuki">Suzuki</option>
                <option value="toyota">Toyota</option>
                <option value="volkswagen">Volkswagon</option>
                    <option value="volvo">Volvo</option>
            </select>
          </div>
          <div class="form-group">
            <label for="vehicleType"><b>Vehicle Type</b></label>
            <select class="form-control" id="vehicleType" name="vehicleType">
                            <option value="bus">Bus</option>
                <option value="combination">Combination</option>
                <option value="convertible">Convertible</option>
                <option value="coupe">Coupe</option>
                <option value="limousine">Limousine</option>
                    <option value="small car">Small Car</option>
                <option value="suv">SUV</option>
                <option value="others">Others</option>
            </select>
          </div>
          <div class="form-group">
            <label for="gearbox_feat"><b>Gear Box Type</b></label>
            <select class="form-control" id="gearbox_feat" name="gearbox_feat">
                <option value="manual">Manual</option>
                <option value="automatic">Automatic</option>
            </select>
          </div>
          <div class="form-group">
            <label for="fuelType_feat"><b>Fuel Type</b></label>
            <select class="form-control" id="fuelType_feat" name="fuelType_feat">
                <option value="cng">CNG</option>
                <option value="diesel">Diesel</option>
                <option value="hybrid">Hybrid</option>
                <option value="lpg">LPG</option>
                <option value="petrol">Petrol</option>
            </select>
          </div>
          <button type="submit" class="btn btn-primary" style = "position:relative;left:500px;top:2px;font-
size:medium;">Predict</button>
          </div>
        </form>
    </div>
</body>
</html>
```

Cars-submit.html (Output Page)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Car Resale Value Prediction</title>
    <link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-
awesome.min.css">

    <style>
        body{
            background-color: #fcde67;
        }
        .container{
            background-color: white;
            margin-top: 10px;
            padding: 20px;
            border-radius: 0px;
        }
        .form-group{
            margin-top: 20px;
        }
        .form-control{
            border-radius: 10px;
        }
        .btn{
            border-radius: 10px;
            margin-top: 20px;
        }
        .fa{
            font-size: 30px;
            color: #f2f2f2;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1 class="text-center">🚙 Obtain the Precise Resale Value of Your Vehicle 🚗</h1><br><br>
        <h3 class="text-center"><i>Predicted Price for Your Car:</i> ₹ {{prediction}}</h3>
    </div>
</body>
</html>
```

Flask Integration with IBM CLOUD:

```
from flask import  Flask,render_template,request
import pickle
import numpy as np

import requests
import json

API_KEY = "HThOlLSFbzwH1uaPHZ2JAZGYK_tAqS5FGQcHd4eGAkQF"
token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":
                                                            API_KEY,
                                                           "grant_type": 'urn:ibm:params:oauth:grant-type:apikey'})
mltoken = token_response.json()["access_token"]

header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

app = Flask(__name__)
model=pickle.load(open("resale_value_pickle_file.sav","rb"))
@app.route("/")
def home():
    return render_template("cars-index.html")

@app.route('/submit',methods=["POST","GET"])
def prediction():
    if request.method=="POST":
        yearofRegistration=request.form["yearofRegistration"]
        monthofRegistration=request.form["monthofRegistration"]
        powerPS = request.form["powerPS"]
        kilometer = request.form["kilometer"]
        notRepairedDamage = request.form["notRepairedDamage"]
        if notRepairedDamage == "Yes":
            notRepairedDamage =1
        elif notRepairedDamage == "No":
            notRepairedDamage =0
        elif notRepairedDamage == "not-declared":
            notRepairedDamage =2



brand = request.form["brand"]
        if brand == "audi":
            brand = 1
        elif brand == "jeep":
            brand = 14
        elif brand == "volkswagen":
            brand = 38
        elif brand == "skoda":
            brand = 31
        elif brand == "bmw":
            brand = 2
```

```
    elif brand == "nissan":
        brand = 23
    elif brand == "renault":
        brand = 27
    elif brand == "ford":
        brand = 10
    elif brand == "honda":
        brand = 11
    elif brand == "mercedes_benz":
        brand = 20
    elif brand == "toyota":
        brand = 36
    elif brand == "hyundai":
        brand = 12
    elif brand == "kia":
        brand = 15
    elif brand == "peugeot":
        brand = 25
    elif brand == "mitsubishi":
        brand = 22
    elif brand == "fiat":
        brand = 9
    elif brand == "volvo":
        brand = 39
    elif brand == "suzuki":
        brand = 35
    elif brand == "porsche":
        brand = 26
    elif brand == "dacia":
        brand = 6



gearbox_feat=request.form["gearbox_feat"]
    if gearbox_feat == "manual":
        gearbox_feat =1
    elif gearbox_feat == "automatic":
        gearbox_feat =0
    fuelType_feat = request.form["fuelType_feat"]
    if fuelType_feat=="petrol":
        fuelType_feat=1
    elif fuelType_feat=="diesel":
        fuelType_feat=3
    elif fuelType_feat=="lpg":
        fuelType_feat=4
    elif fuelType_feat=="hybrid":
        fuelType_feat=6
    elif fuelType_feat=="cng":
        fuelType_feat=7
    vehicleType = request.form["vehicleType"]
    if vehicleType == "coupe":
        vehicleType = 3
```

```
        elif vehicleType == "suv":
            vehicleType = 8
        elif vehicleType == "small car":
            vehicleType = 7
        elif vehicleType == "limousine":
            vehicleType = 4
        elif vehicleType == "bus":
            vehicleType = 0
        elif vehicleType == "combination":
            vehicleType = 1
        elif vehicleType == "others":
            vehicleType = 6
        elif vehicleType == "convertible":
            vehicleType = 7


        int_features = [yearofRegistration,
powerPS,kilometer,monthofRegistration,notRepairedDamage,brand,gearbox_feat,fuelType_feat,vehicleTyp
e]
        features = [np.array(int_features, dtype=int)]




payload_scoring = {"input_data": [{"field": [
            ["price", 'vehicleType', 'yearOfRegistration', 'gearbox', 'powerPS', 'model', 'kilometer',
             'monthOfRegistration', 'fuelType', 'brand', 'notRepairedDamage']],
                                    "values": features }]}
        response_scoring = requests.post(
            'https://us-south.ml.cloud.ibm.com/ml/v4/deployments/9aaee049-3c55-47c0-ae13-
206286158edf/predictions?version=2022-11-18',
            json=payload_scoring,
            headers={'Authorization': 'Bearer ' + mltoken})
        print("Scoring response")
        print(response_scoring.json())
        predictions = response_scoring.json()
        prediction=model.predict(features)
        return render_template("cars-submit.html",prediction=round(prediction[0],2))

if __name__=="__main__":
    app.run(debug=True)
```

15. GITHUB LINK:

https://github.com/IBM-EPBL/IBM-Project-13083-1659509975

16. PROJECT DEMO VIDEO LINK:

https://vimeo.com/772816678