**Team No : PNT2022TMID33983**

1. **INTRODUCTION**
   1. **Project Overview :**

Determining whether the listed price of a used car is a challenging task, due to the many factors that drive a used vehicle’s price on the market. The focus of this project is developing machine learning models that can accurately predict the price of a used car based on its features, in order to make informed purchases. We implement and evaluate various learning methods on a dataset consisting of the sale prices of different makes and models . We will compare the performance of various machine learning algorithms like Linear Regression, Ridge Regression, Lasso Regression, Elastic Net, Decision Tree Regression and choose the best out of it. Depending on various parameters we will determine the price of the car. Regression Algorithms are used because they provide us with continuous value as an output and not a categorized value because of which it will be possible to predict the actual price a car rather than the price range of a car. User Interface has also been developed which acquires input from any user and displays the Price of a car according to user’s inputs.

* 1. **Purpose :**

With difficult economic conditions, it is likely that sales of secondhand imported (reconditioned) cars and used cars will increase. In many developed countries, it is common to lease a car rather than buying it outright. After the lease period is over, the buyer has the possibility to buy the car at its residual value, i.e. its expected resale value. Thus, it is of commercial interest to sellers/financiers to be able to predict the salvage value (residual value) of cars with accuracy.

In order to predict the resale value of the car, we proposed an intelligent, flexible, and effective system that is based on using regression algorithms. Considering the main factors which would affect the resale value of a vehicle a regression model is to be built that would give the nearest resale value of the vehicle. We will be using various regression algorithms and algorithm with the best accuracy will be taken as a solution, then it will be integrated to the web-based application where the user is notified with the status of his product.

1. **LITERATURE SURVEY**
   1. **Existing Problem:**

In existing problem, the solution is not good to predict the car prize. To predict the used car value , we need to concentrate in more accuracy by using algorithms to give accuracy. In this solution does not predict correct value of the used car.

* 1. **References:**
     1. Sameerchand Pudaruth, “Predicting the Price of Used Cars using Machine Learning Techniques”;(IJICT 2014)
     2. Enis gegic, Becir Isakovic, Dino Keco, Zerina Masetic, Jasmin Kevric, ”Car Price Prediction Using Machine Learning”; (TEM Journal 2019)
     3. Ning sun, Hongxi Bai, Yuxia Geng, Huizhu Shi, “Price Evaluation Model In Second Hand Car System Based On BP Neural Network Theory”;

(Hohai University Changzhou, China)

* + 1. Nitis Monburinon, Prajak Chertchom, Thongchai Kaewkiriya, Suwat Rungpheung, Sabir Buya, Pitchayakit Boonpou, “Prediction of Prices for Used Car by using Regression Models” (ICBIR 2018) [5] 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).
  1. **Problem Statement:**

The main aim of this project is to predict the price of used cars using the various Machine Learning (ML) models. This can enable the customers to make decisions based on different inputs or factors namely

* Brand or Type of the car one prefers like Ford, Hyundai

* Model of the car namely Ford Figo, Hyundai Creta

* Location like Delhi, Chennai, Mumbai

* Year of manufacturing like 2020, 2021

* Type of fuel namely Petrol, Diesel

* Price range or Budget

* Type of transmission which the customer prefers like Automatic or

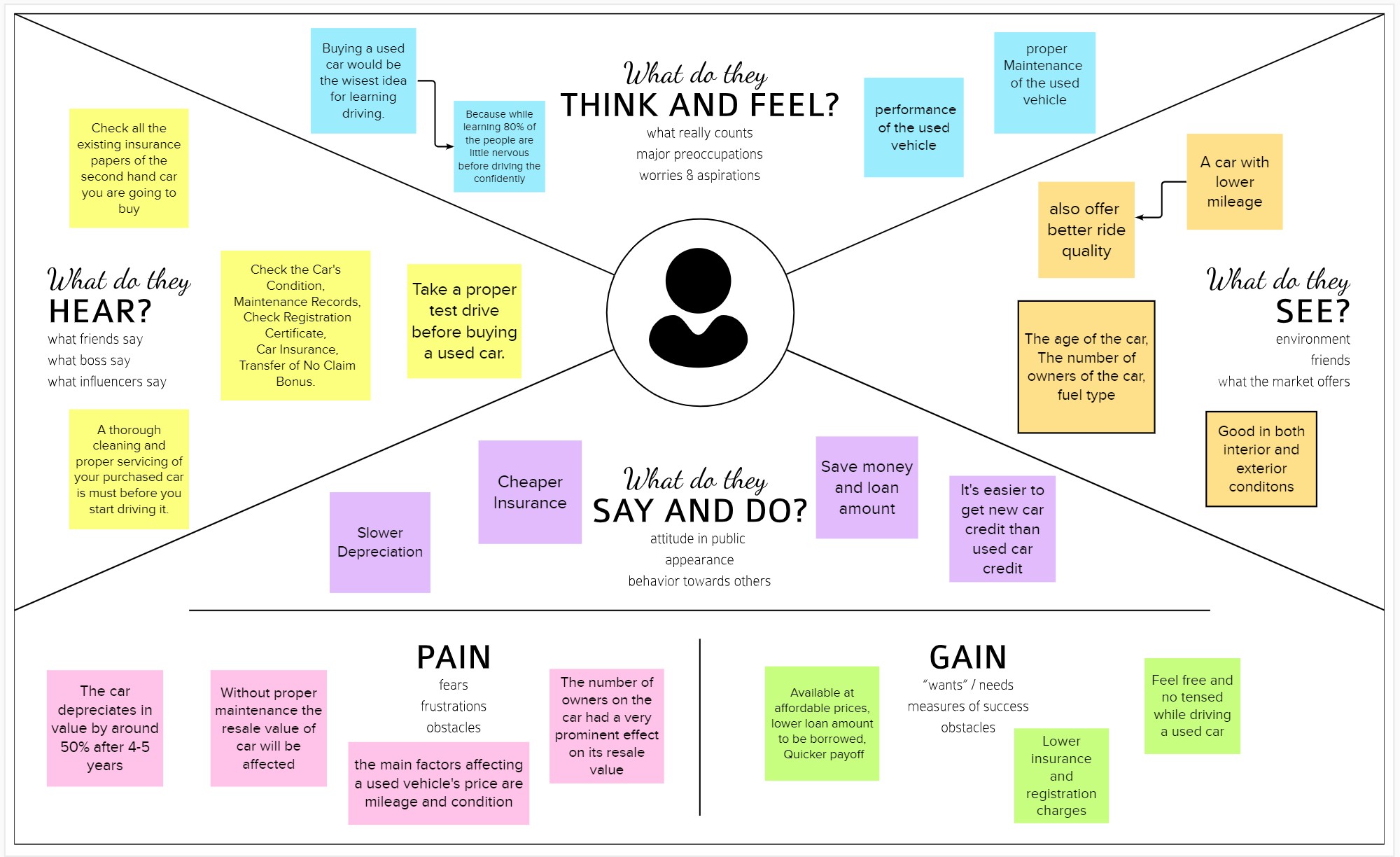
Manual

* Mileage

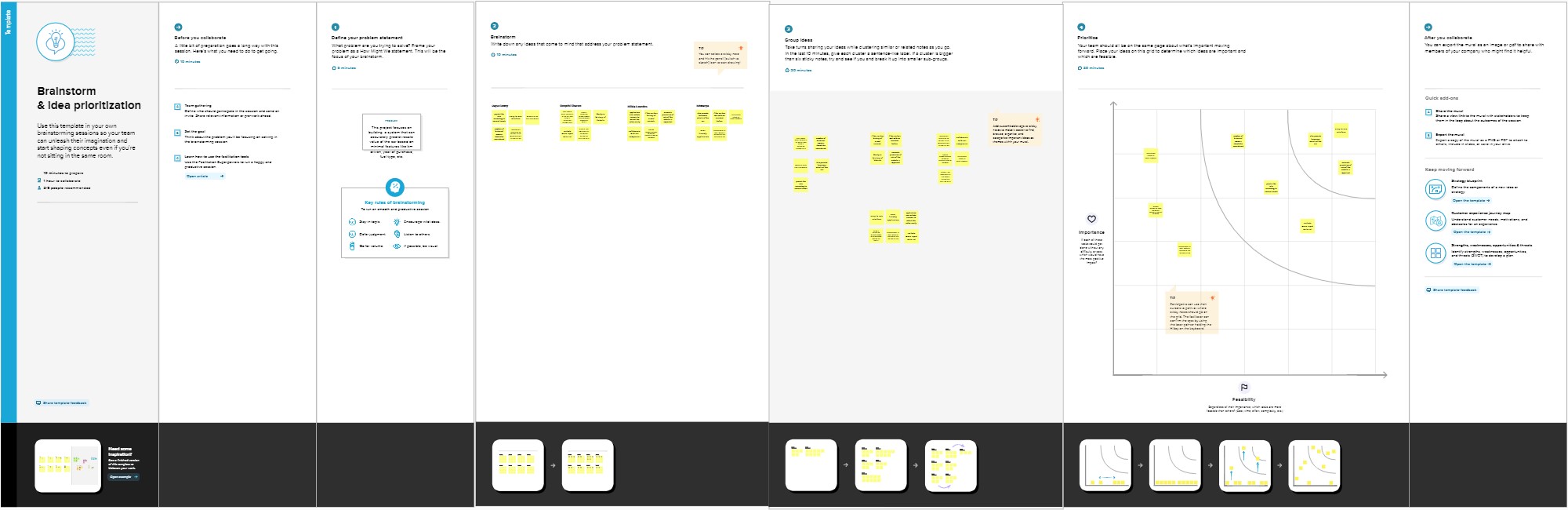
to name a few characteristic features required by the customer. The project Car Price Prediction deals with providing the solution to these problems. Through this project, we will get to know which of the factors are significant and tell us how they affect the car’s worth in the market

**3.Ideation and Proposed Solution**

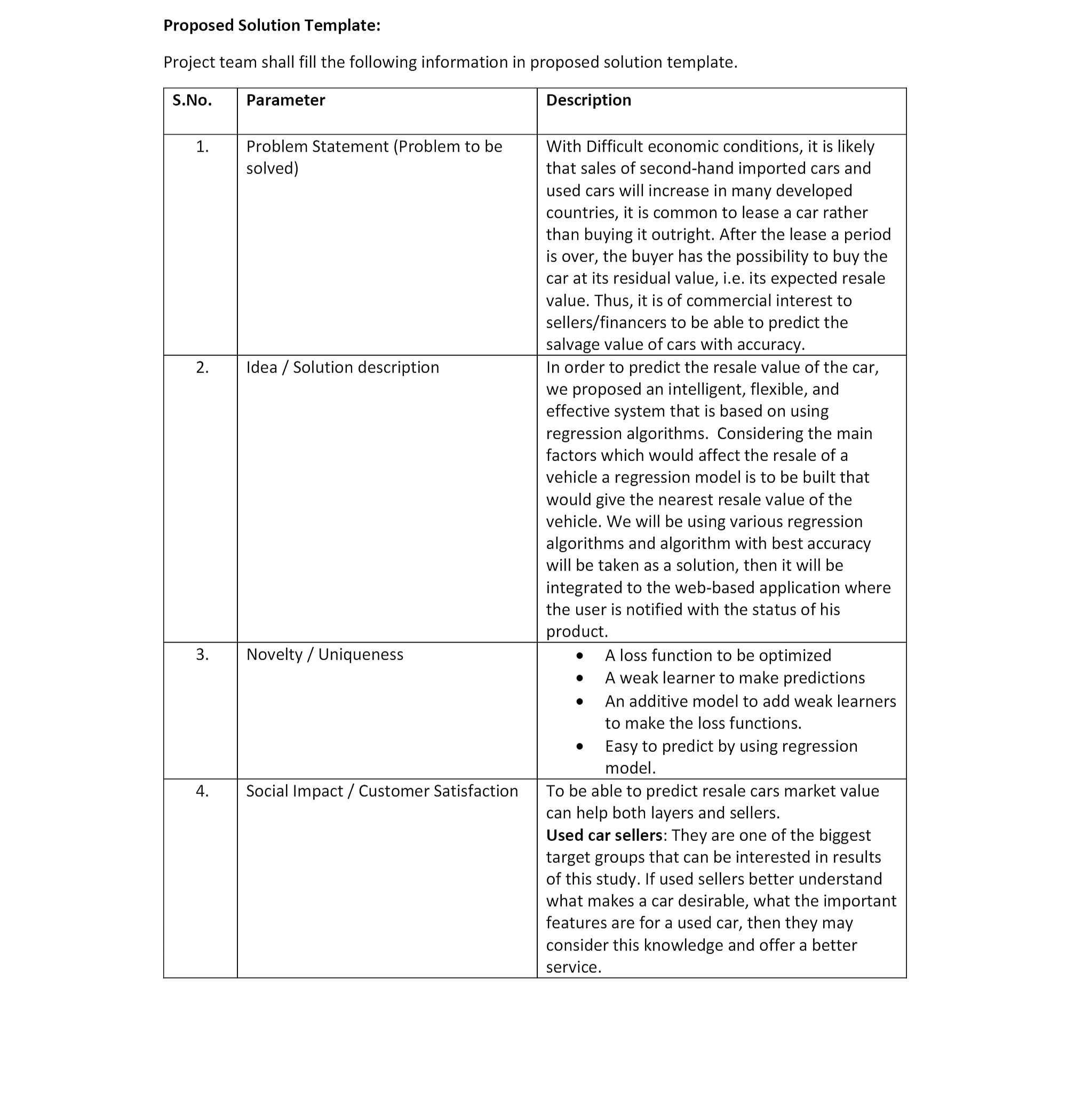
* 1. **Empathy Map Canvas:**



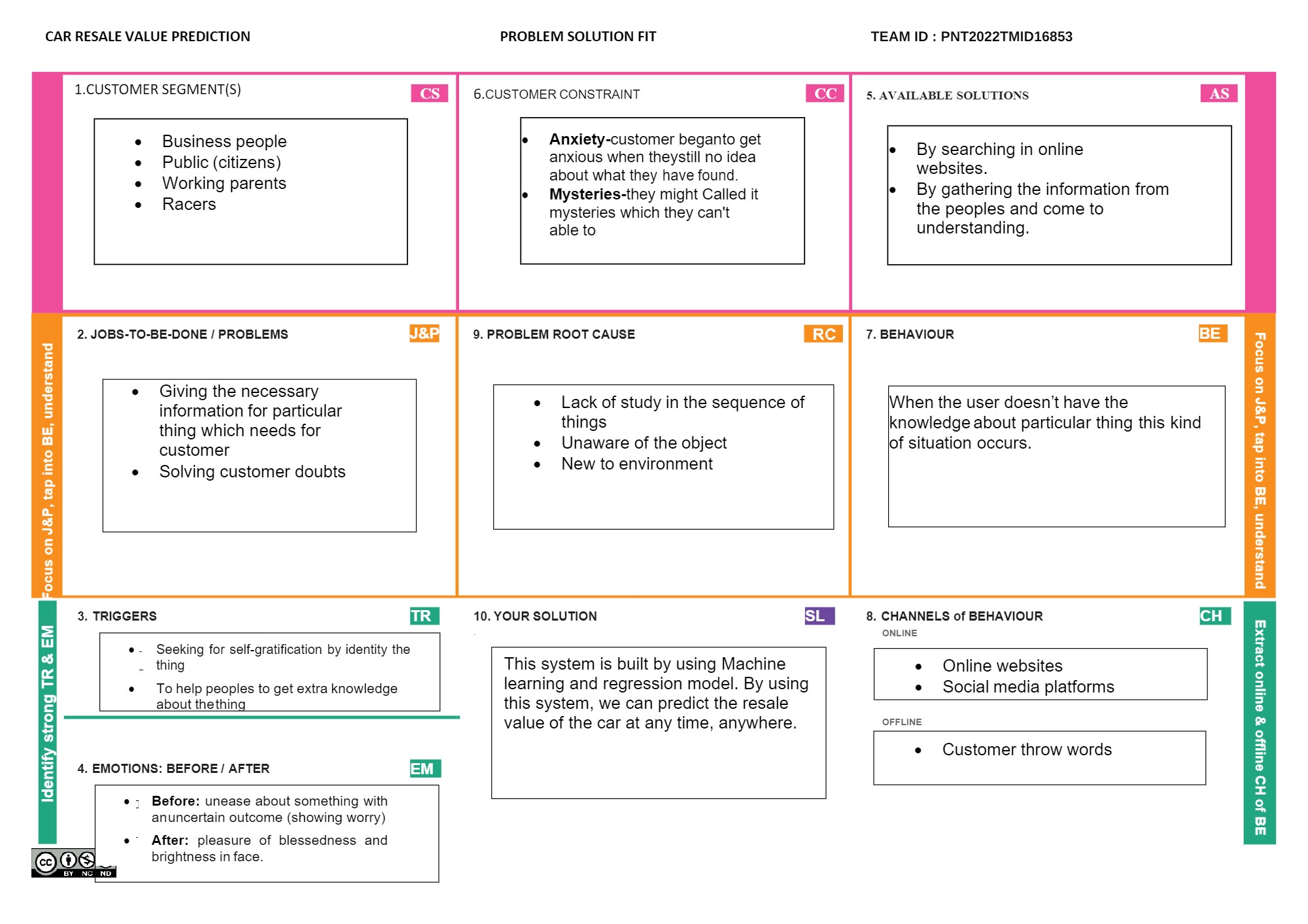
* 1. **Ideation And Brainstorming:**



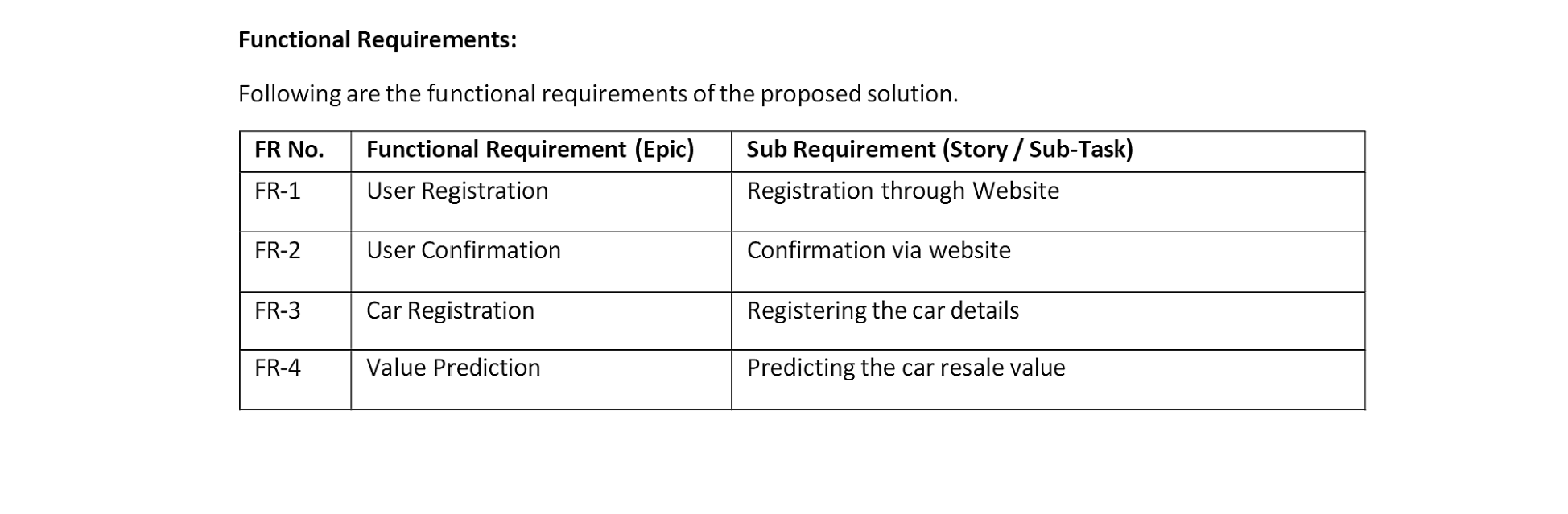
* 1. **Proposed Solution:**



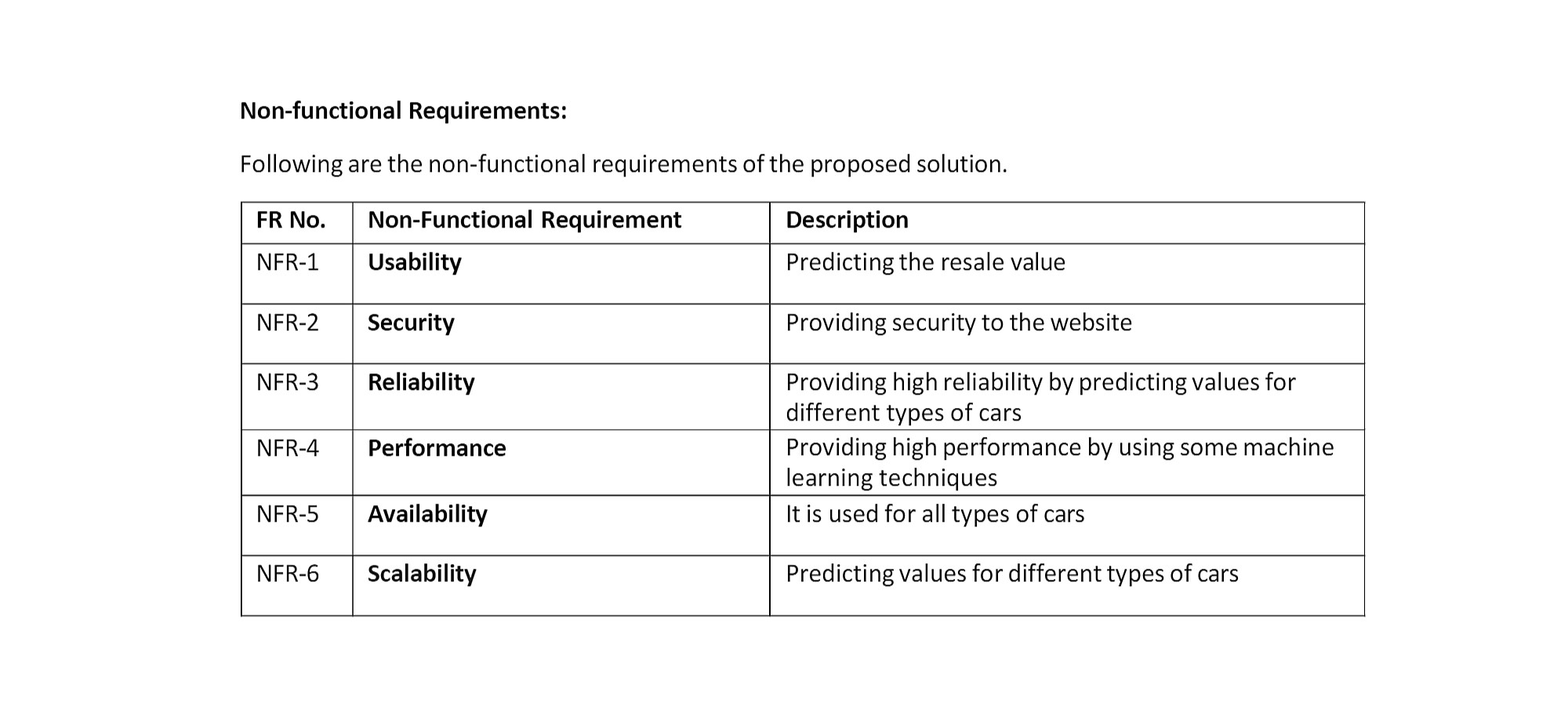
* 1. **Problem Solution Fit:**



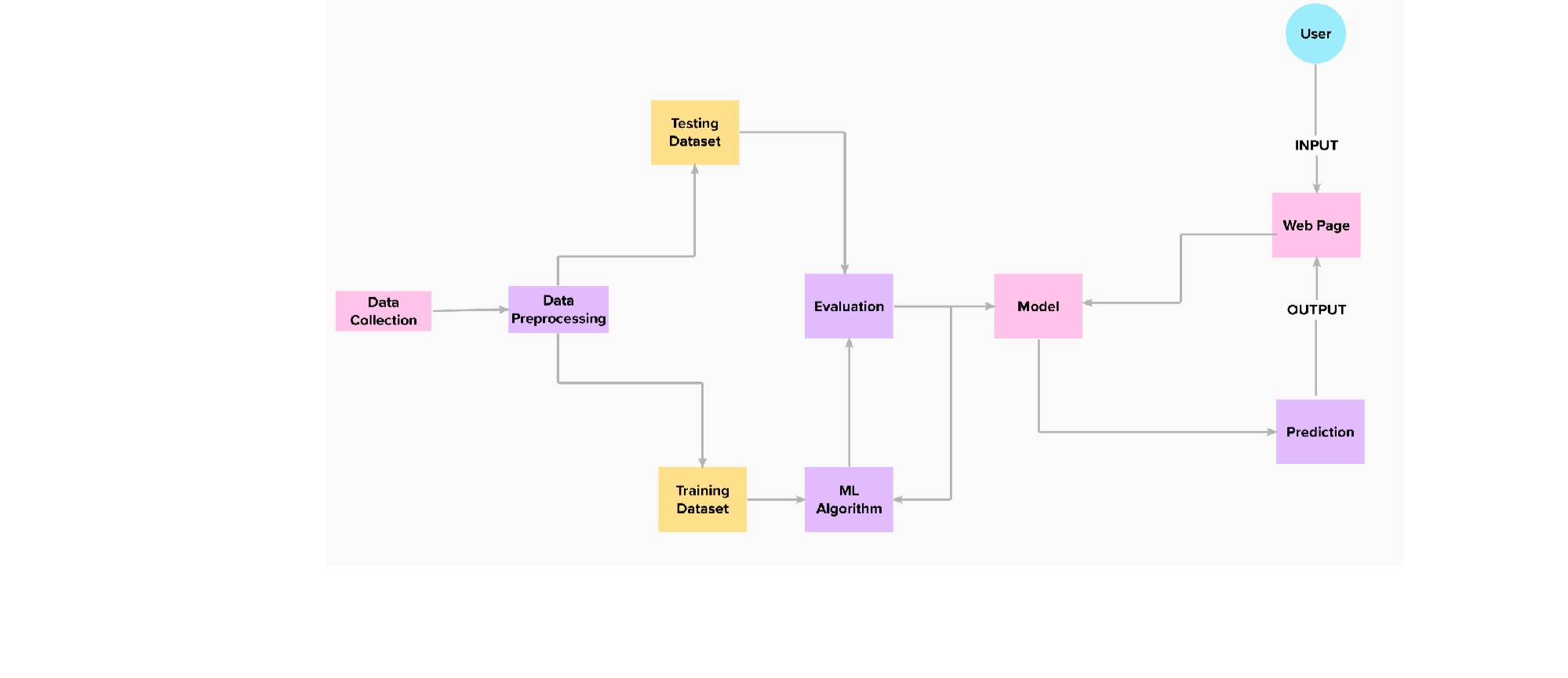
1. **Requirements Analysis:**
   1. **Functional Requirements:**



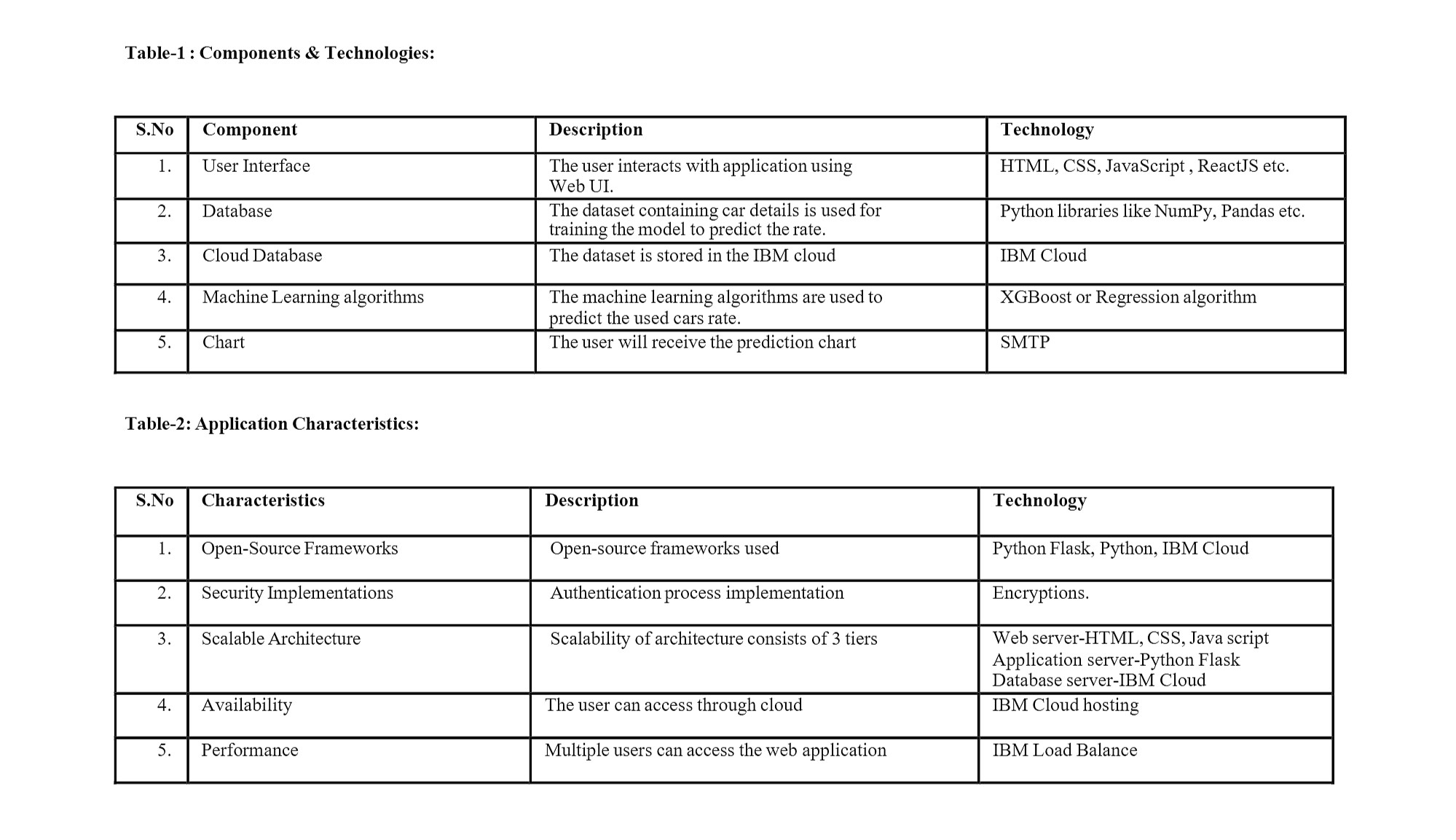
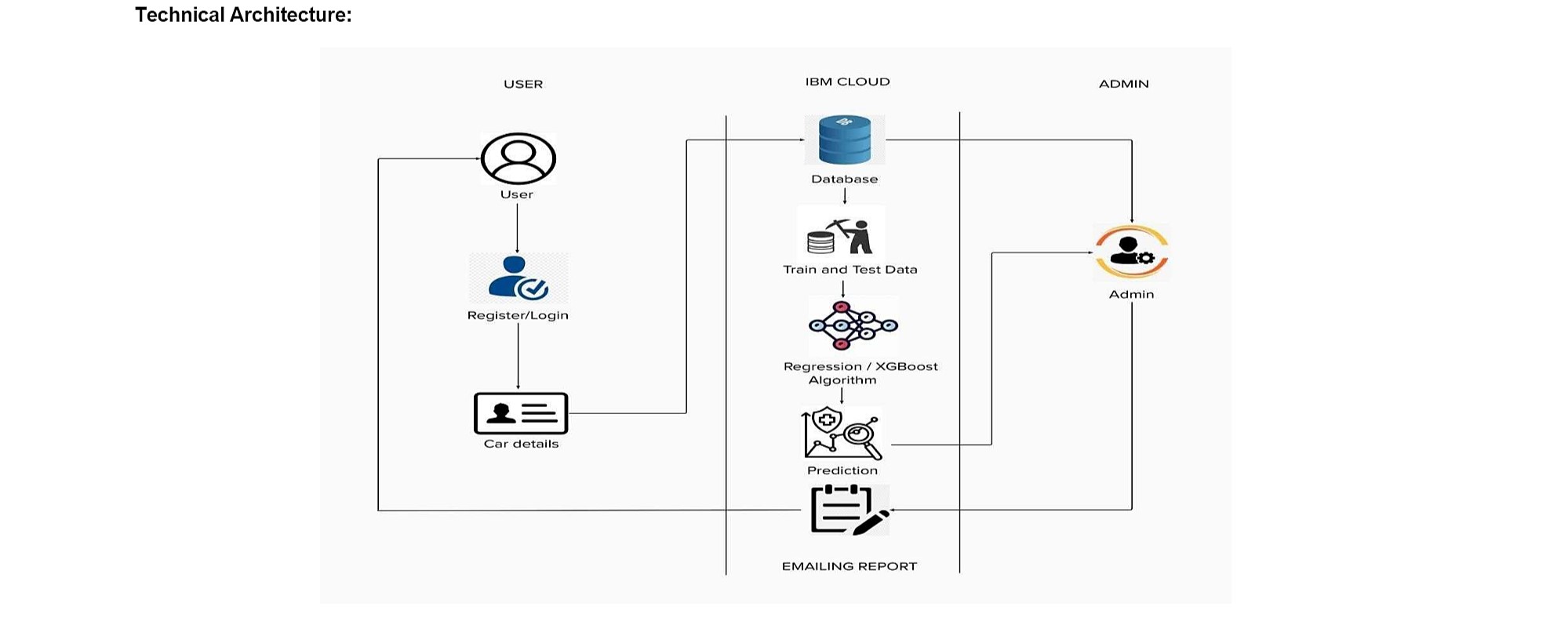
* 1. **Non Functional Requirements:**



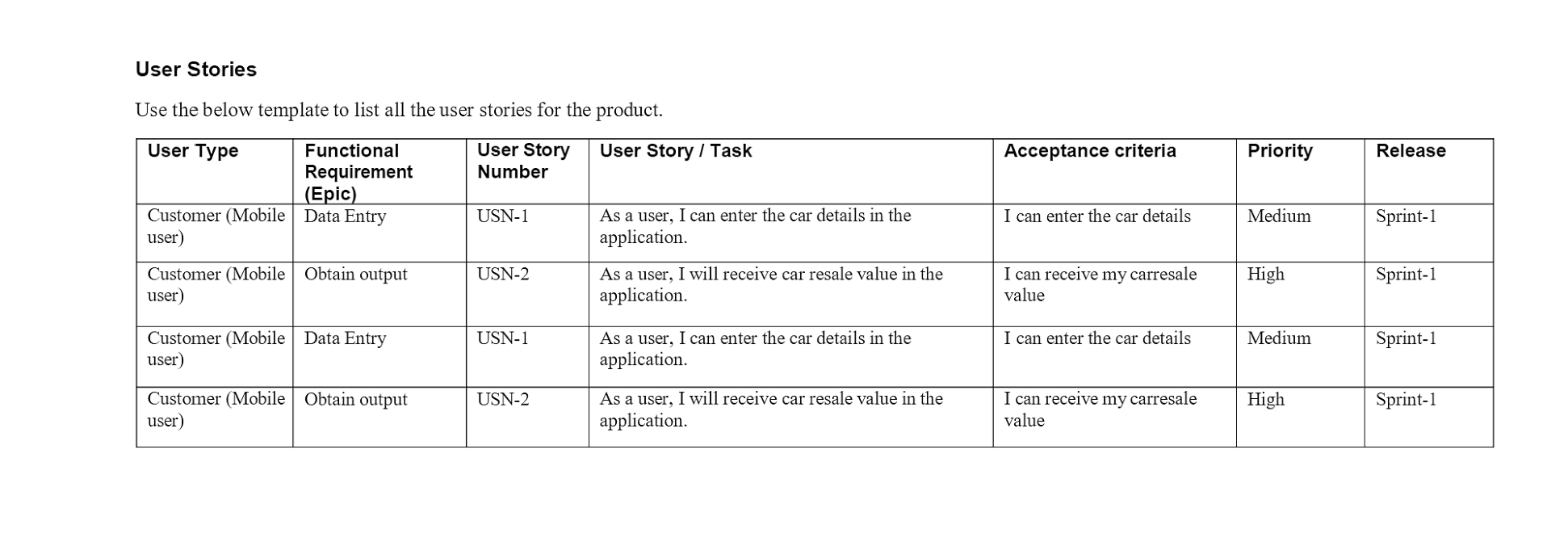
1. **Project Design:**
   1. **Data Flow Diagram:**



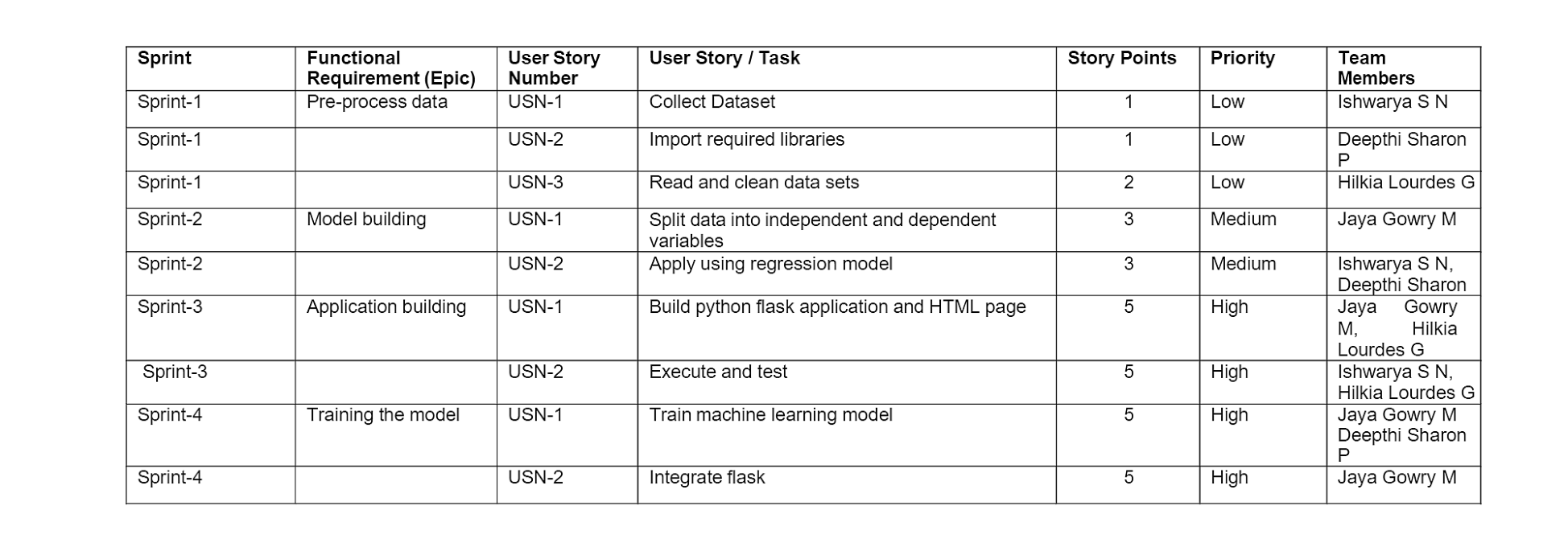
* 1. **Solution and Technical Architecture:**



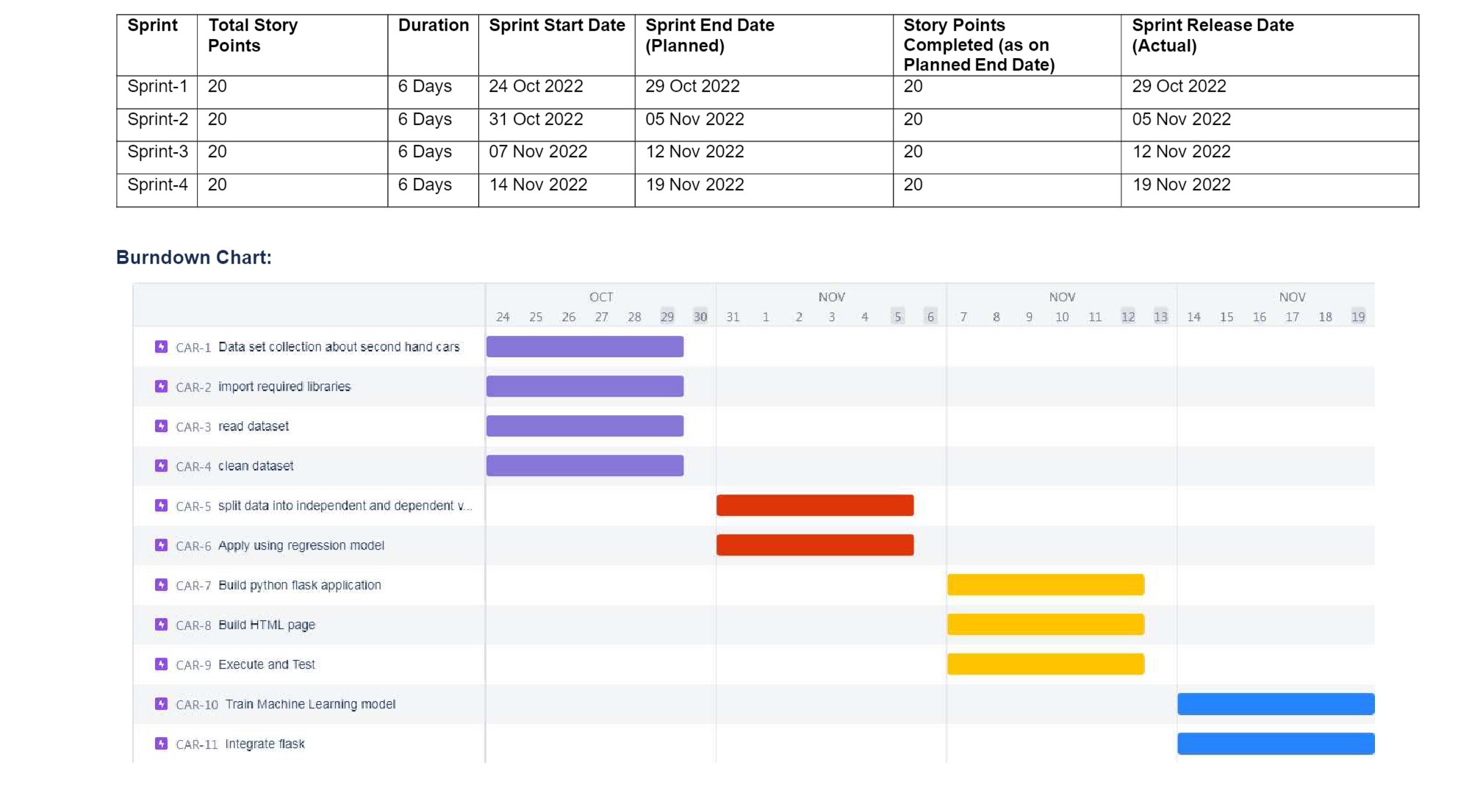
* 1. **User Stories:**



1. **Project planning and scheduling:**
   1. **Sprint Planning and estimation:**



1. **2.Sprint Delivery Schedule:**



1. **Coding and Solutioning:**

**CSS CODE for predict**

.header{

min-height: 100vh;

width: 100%;

background-image: lineargradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car6.png);

background-position: center;

background-size: cover;

position: relative;

}

.text-box{

text-align: center;

position: relative;

color: #FFE4C4;

top:50%;

}

.text-box h1{

margin-top: 50px;

font-size: 55px;

}

.text-box p{

margin: 10px 0 40px;

font-size: 15px;

}

body{ margin: 0;

}

nav{

display:flex;

padding: 2% 6%;

justify-content: space-between;

align-items: center;

}

**CSS code for style**

\*{

margin: 0;

padding: 0;

}

.header{

min-height: 100vh;

width: 100%;

background-image: linear-

gradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car1.png);

background-position: center;

background-size: cover;

position: relative;

}

nav{

display:flex;

padding: 2% 6%;

justify-content: space-between;

align-items: center;

}

.nav-links{

flex: 1;

text-align: right;

}

.nav-links ul li{

list-style: none;

display: inline-block;

padding: 8px 12px;

position: relative;

}

.nav-links ul li a{

color:white;

text-decoration: none;

font-size: 13px;

}

.text-box{

text-align: center;

position: relative;

color: #FFE4C4;

top:50%;

.text-box h1{

margin-top: 50px;

font-size: 55px;

} .text-box p{

margin: 10px 0 40px;

font-size: 15px;

} .visit-btn{

display: inline;

border: 3px solid #fff;

padding:10px 14px;

font-size: 15px;

background: transparent;

color: white;

text-decoration:none;

}

**CSS code for value**

.header{

width: 100%;

text-align: center;

//padding-top: 20px;

font-size:20px;

font-family: "Lucida Console";

background-color:#43FFB6;

border:0%;

top:0px;

bottom:0px;

right:0px;

left:0px;

overflow-y:auto;

body{

margin: 0;

}

.form{

background-image: lineargradient(rgba(25,30,30,0.7),rgba(25,30,30,0.7)),url(../Images/car4.jpg);

background-position: center;

background-size: cover;

position: relative;

}

.form{ text-align: center; padding:20px; text-top:10px; display: flex; flex-direction: column;

align-items: center;

}

.form{ font-size:22px;

}

textarea { width: 100%;

height: 150px;

padding: 12px 20px;

box-sizing: border-box;

border: 2px solid #ccc;

border-radius: 4px;

background-color: #f8f8f8;

resize: none;

input[type=text] {

transition: width 0.4s ease-in-out;

}

input[type=text] { width: 70%;

height: 10%;

padding: 10px 10px;

margin: 5px 0;

}

#model{

width: 70%;

}

#brand{ width:70%;

}

#vehicle{ width:70%;

} \*{

color:black;

}

#button{

padding: 10px 10px;

margin: 0;

text-align:center;

width:100px;

}

**Build an Html Page** **car.html** <!DOCTYPE html>

<html lang="en" dir="ltr">

<head>

<meta charset="utf-8">

<title>Car resale value </title>

<link rel="stylesheet" href="../static/css/style.css">

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/fontawesome/4.7.0/css/font-awesome.min.css">

</head>

<body>

<section class="header">

<nav>

<a href="/"><img src="../static/Images/sang.png" width="100" height="100"></a>

</nav>

<div class="text-box">

<h1>Car resale value Predictor</h1>

<p>Best system to predict the amount of resale value based on the parameters provided by the user .</p>

<a href="./predict\_page" class="visit-btn ">Check price</a>

</div>

</section>

</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">

<link rel="stylesheet" href="../static/css/predict.css">

<title>Car Resale Predicted Value</title>

</head> <body>

<section class="header">

<nav>

<a href="/"><img src="../static/Images/sang.png" width="100" height="100"></a>

</nav>

<div class="text-box">

<h1>The Predicted Car Resale Value is </h1>

<h1>{{predict}}</h1>

</div>

</section>

</body>

</html>

**Build a python flask app:** # Import Libraries import pandas as pd import numpy as np

from flask import Flask, render\_template, Response, request import pickle

from sklearn.preprocessing import LabelEncoder import requests

# NOTE: you must manually set API\_KEY below using information retrieved from your IBM Cloud account.

API\_KEY = "Qo9j8ni7qMJ8j1C8VFDRFHbuGRAhYWcTlkVqnYg1AGkE" token\_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":API\_KEY, "grant\_type": 'urn:ibm:params:oauth:granttype:apikey'})

mltoken = token\_response.json()["access\_token"]

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

app = Flask(\_\_name\_\_)#initiate flask app

def load\_model(file='../Result/resale\_model.sav'):#load the saved model return pickle.load(open(file, 'rb'))

@app.route('/') def index():#main page return render\_template('car.html')

@app.route('/predict\_page') def predict\_page():#predicting page return render\_template('value.html')

@app.route('/predict', methods=['GET','POST']) def predict(): reg\_year = int(request.args.get('regyear')) powerps = float(request.args.get('powerps')) kms= float(request.args.get('kms')) reg\_month = int(request.args.get('regmonth'))

gearbox = request.args.get('geartype') damage = request.args.get('damage') model = request.args.get('model') brand = request.args.get('brand') fuel\_type = request.args.get('fuelType') veh\_type = request.args.get('vehicletype')

new\_row = {'yearOfReg':reg\_year, 'powerPS':powerps,

'kilometer':kms,

'monthOfRegistration':reg\_month,

'gearbox':gearbox,

'notRepairedDamage':damage,

'model':model, 'brand':brand,

'fuelType':fuel\_type,

'vehicletype':veh\_type}

print(new\_row)

new\_df =

pd.DataFrame(columns=['vehicletype','yearOfReg','gearbox',

'powerPS','model','kilometer','monthOfRegistration','fuelType',

'brand','notRepairedDamage']) new\_df = new\_df.append(new\_row, ignore\_index=True) labels =

['gearbox','notRepairedDamage','model','brand','fuelType','vehicletype'] mapper = {}

for i in labels: mapper[i] = LabelEncoder() mapper[i].classes =

np.load('../Result/'+str('classes'+i+'.npy'), allow\_pickle=True) transform = mapper[i].fit\_transform(new\_df[i]) new\_df.loc[:,i+'\_labels'] = pd.Series(transform,

index=new\_df.index)

labeled = new\_df[['yearOfReg','powerPS','kilometer','monthOfRegistration'] +

[x+'\_labels' for x in labels]]

X = labeled.values.tolist()

print('\n\n', X)

#predict = reg\_model.predict(X)

# NOTE: manually define and pass the array(s) of values to be

scored in the next line payload\_scoring = {"input\_data": [{"fields": [['yearOfReg',

'powerPS', 'kilometer', 'monthOfRegistration','gearbox\_labels',

'notRepairedDamage\_labels', 'model\_labels','brand\_labels', 'fuelType\_labels',

'vehicletype\_labels']], "values": X}]}

response\_scoring = requests.post('https://us-

south.ml.cloud.ibm.com/ml/v4/deployments/7f67cbed-6222-413b-9901b2a72807ac82/predictions?version=2022-10-30', json=payload\_scoring,

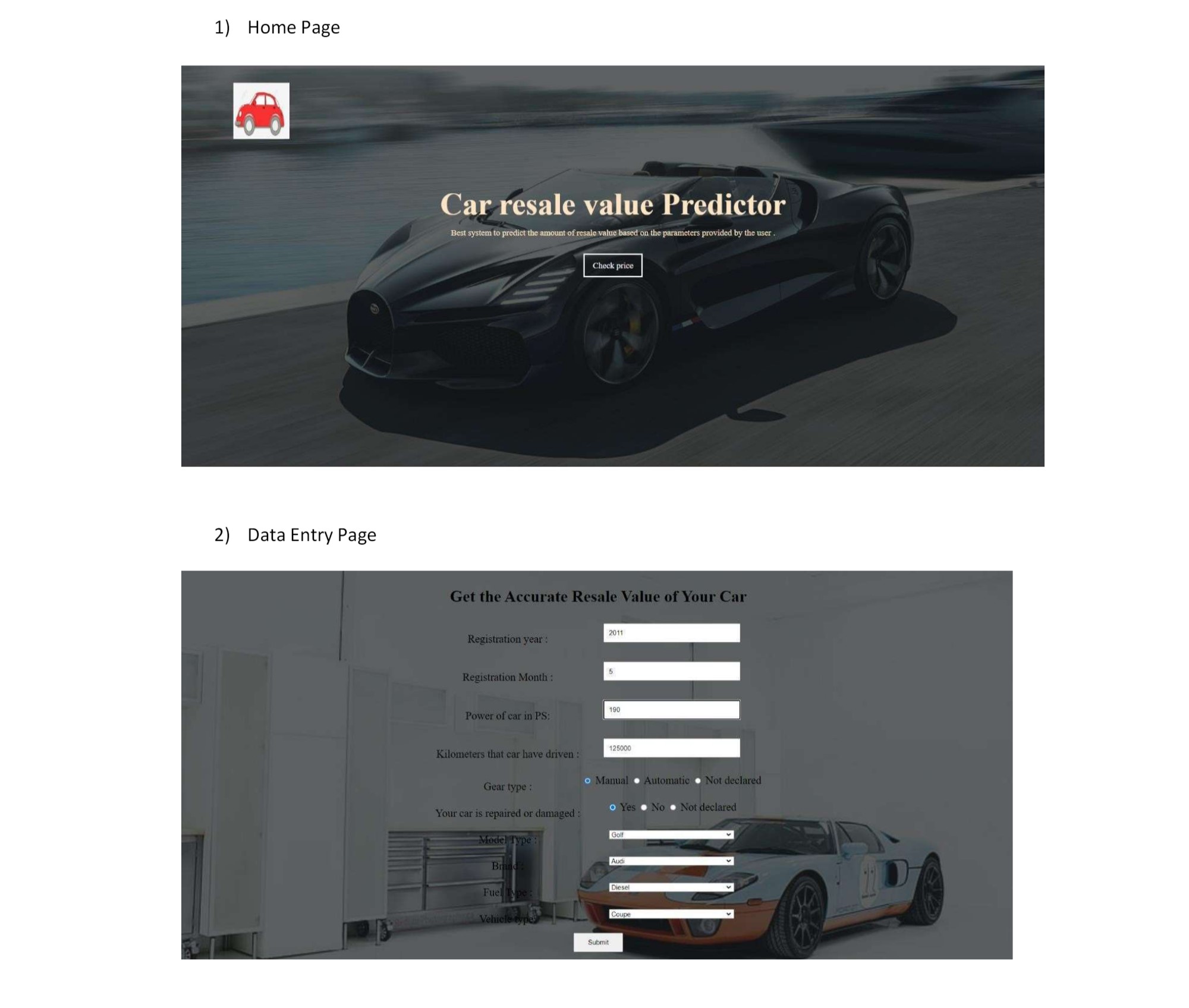
headers={'Authorization': 'Bearer ' + mltoken}) predictions = response\_scoring.json() print(response\_scoring.json())

predict = predictions['predictions'][0]['values'][0][0] print("Final prediction :",predict)

return render\_template('predict.html',predict=predict)

if \_\_name\_\_=='\_\_main\_\_': reg\_model = load\_model()#load the saved model app.run(host='localhost', debug=True, threaded=False)

1. **Testing:**



1. **RESULT**

**Car Resale Valur Display:**



1. **Advantages And disadvantages:**

**Advantages:**

* + Good at learning complex and non linear relationships
  + highly explainable and easy to interpret
  + Robust to outliers
  + No features scaling is required
  + Easy to predict the current rate of car

**Disadvantages:**

* + consumes more time
  + Require high computational power
  + require more data to predict exact value and accuracy

1. **Conclusion:**

The increased prices of new cars and the financial incapability of the customers to buy them, Used Car sales are on a global increase. Therefore, there is an urgent need for a Used Car Price Prediction system which effectively determines the worthiness of the car using a variety of features. The proposed system will help to determine the accurate price of used car price prediction. This paper compares 3 different algorithms for machine learning : Linear Regression, Lasso and Ridge Regression.

1. **Future Scope:**

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.

1. **Appendix:** **Github Link:** https://github.com/IBM-EPBL/IBM-Project-34693-1660271756 **Weights and basis link:** https://wandb.ai/jayagowry/car\_resale\_value?workspace=user-jayagowry **Google colab link for run the model:**

https://colab.research.google.com/drive/10EHwGiA\_KNKygUsjIA0G0Sr3YS

9GAcXW?usp=sharing

https://colab.research.google.com/drive/1umCW77LeW3srx\_wW\_0SVXllWa0jDdLL?usp=sharing