CAR RESALE VALUE PREDICTION

IBM Naalaiya Thiran

A PROJECT REPORT BY

TEAM ID: PNT2022TMID35625

SUDHARSAN B - 2019115109

VISWANATH PATHMANABAN - 2019115124

SHENBAGA SHUDHAN VRR - 2019115130

RUKESH A - 20191151084

1. INTRODUCTION

This project, "Car Resale Value Prediction," intends to create a model that predicts used car values based on a variety of factors such as vehicle miles, year of manufacture, fuel consumption, transmission, road tax, fuel type, and engine capacity. In the used automobile market, this strategy may benefit vendors, purchasers, and car manufacturers. Upon completion, it may generate a somewhat accurate price prediction based on the data entered by users. Machine learning and data science are used in the model-building process. The dataset was obtained from used car listings. To attain the maximum accuracy, the research used a variety of regression approaches, including linear regression, polynomial regression, support vector regression, decision tree regression, and random forest regression.This study visualized the data before beginning model-building to better comprehend the dataset. The dataset was partitioned and changed to match the regression, assuring the model's performance.

1.1. Project Overview

A car price prediction has been a high interest research area, as it requires noticeable effort and knowledge of the field expert. Considerable number of distinct attributes are examined for the reliable and accurate prediction. To build a model for predicting the price of used cars, the applied three machine learning techniques are random forest ,KN-N and linear regression algorithm. Respective performances of different algorithms were then compared to find one that best suits the 4 available dataset. This ability to capture data, analyze it and use it to personalize a shopping experience or implement is the future of retail.

Parameters involved :

Car name;Year;Selling Price;Present Price;Kms Driven;Fuel type; Seller type;Transmission;Owner

1.2. Purpose

Car resale value prediction assists the user in predicting the resale value of a car based on numerous factors such as kilometers driven, fuel type, and so on. This resale value prediction method is designed for general usage, predicting the amount that the user may roughly buy. The most important criteria for predicting are the vehicle's brand and model, period of usage, mileage, gear type, and fuel type used in the vehicle, since fuel consumption per mile has a significant impact on the cost of a vehicle due to continual variations in the cost of a fuel. The vehicle value projection has been done exactly in light of the many features and elements, as well as with the aid of master data.

2. LITERATURE SURVEY

2.1. Existing problem

The problem is defined as the most efficient method of estimating insurance costs depending on the manufacturer, with some additional expenses borne by the government in the form of taxes. Because the present techniques for calculating the cost require a lot of time and energy, and because of the rising price of new automobiles and consumers' inability to acquire new cars due to a lack of cash, used car sales are on the rise globally.

Customers buying a new automobile may be confident that the money they invest will be worthwhile since the industry sets the costs of new cars.A used car price prediction system is required to efficiently assess the worthiness of the car using a range of factors. There are websites that provide this service,

however their prediction approach may not be the best. Furthermore, several models and techniques may contribute to the ability to anticipate the real market value of a used car. It is critical to understand their true market value while purchasing and selling.

2.2. References

At present, under the guidance of the new generation of information technology,the rapid accumulation of data, the continuous improvement of computing power, the continuous optimization of algorithm models, and the rapid rise of multi-scene applications have made profound changes in the development environment of Machine Learning.

1. S, K. (2020). Prediction of Resale Value of the Car Using Linear Regression Algorithm. In the International Journal of Innovative Science and Research Technology (Vol. 5, Issue 7, pp. 382–386). International Journal of Innovative Science and Research Technology. https://doi.org/10.38124/ijisrt20jul388

2. Used Cars Price Prediction using Supervised Learning Techniques. (2019). In International Journal of Engineering and Advanced Technology (Vol. 9, Issue 1S3, pp. 216–223). Blue Eyes Intelligence Engineering and Sciences Engineering and Sciences Publication - BEIESP. https://doi.org/10.35940/ijeat.a1042.1291s319

3. Hankar, M., Birjali, M., & Beni-Hssane, A. (2022). Used Car Price Prediction using Machine Learning: A Case Study. In 2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC). 2022 11th International Symposium on Signal, Image, Video and Communications (ISIVC). IEEE. https://doi.org/10.1109/isivc54825.2022.9800719

4. Narayana, C. V., Likhitha, C. L., Bademiya, S., & Kusumanjali, K. (2021). Machine Learning Techniques To Predict The Price Of Used Cars: Predictive Analytics in Retail Business. In 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC). 2021 Second International Conference on Electronics and Sustainable Communication Systems (ICESC). IEEE. https://doi.org/10.1109/icesc51422.2021.9532845

5. Jin, C. (2021). Price Prediction of Used Cars Using Machine Learning. In 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT). 2021 IEEE International Conference on Emergency Science and Information Technology (ICESIT). IEEE. https://doi.org/10.1109/icesit53460.2021.9696839

6. Çelik, Ö., & Osmanoğlu, U. Ö. (2019). İkinci El Araba Fiyatlarının Tahmini. In European Journal of Science and Technology (pp. 77–83). European Journal of Science and Technology. https://doi.org/10.31590/ejosat.542884

7. Gegic, Enis, et al. "Car price prediction using machine learning techniques." TEM Journal 8.1 (2019): 113.https://doi.org/10.18421/TEM81-16

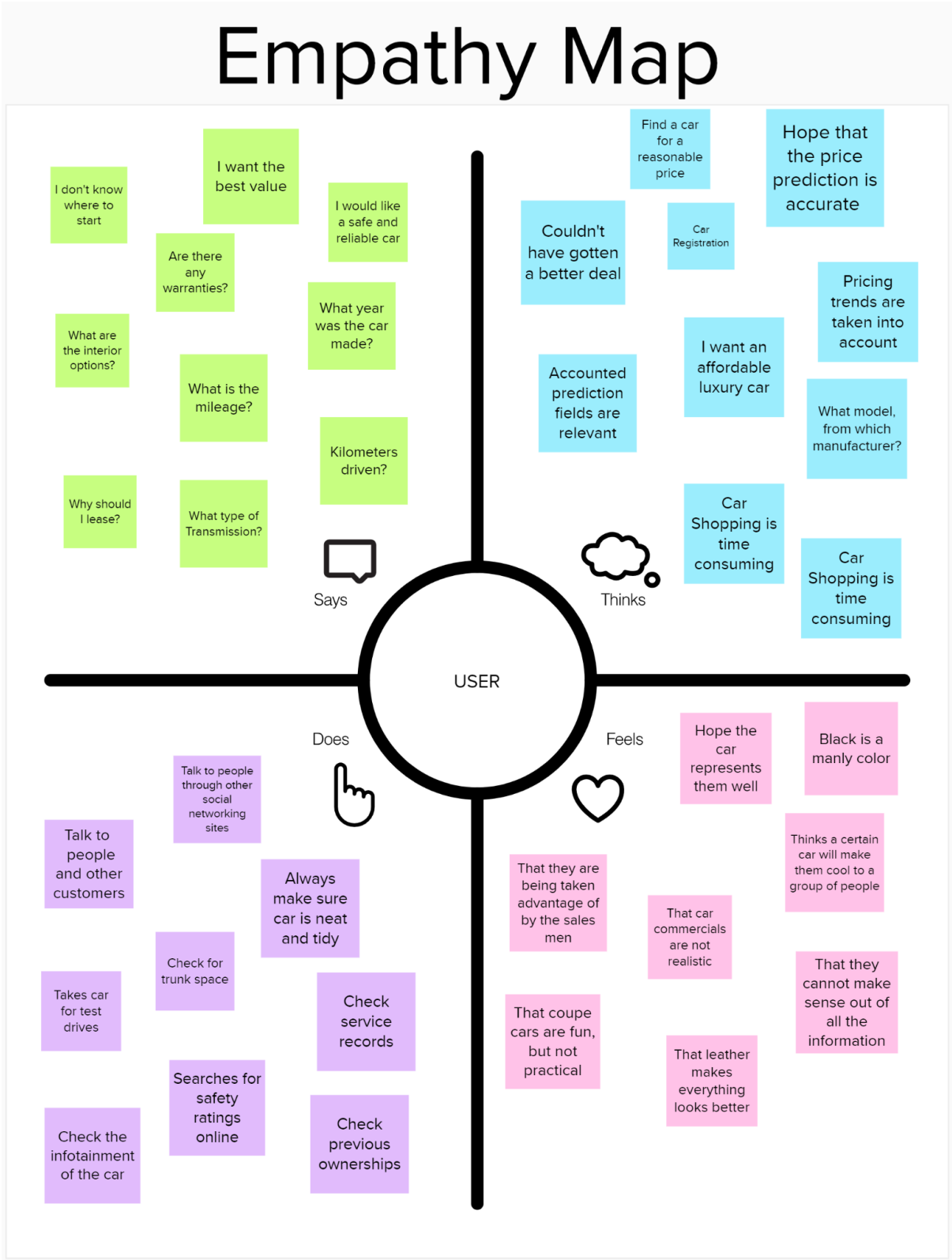
8. Chang, J., & Zhang, Y. (2018). Research on Second-hand Vehicle Evaluation System Based on Improved Replacement Cost Method. In Proceedings of the 2018 7th International Conference on Energy, Environment and Sustainable Development (ICEESD 2018). 2018 7th International Conference on Energy, Environment and Sustainable Development (ICEESD 2018). Atlantis Press. https://doi.org/10.2991/iceesd-18.2018.23

2.3. Problem Statement Definition

To develop an intelligent, adaptable and efficient model using regression methods to give an accurate prediction of resale value of cars and build and integrate a web-application with it.

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

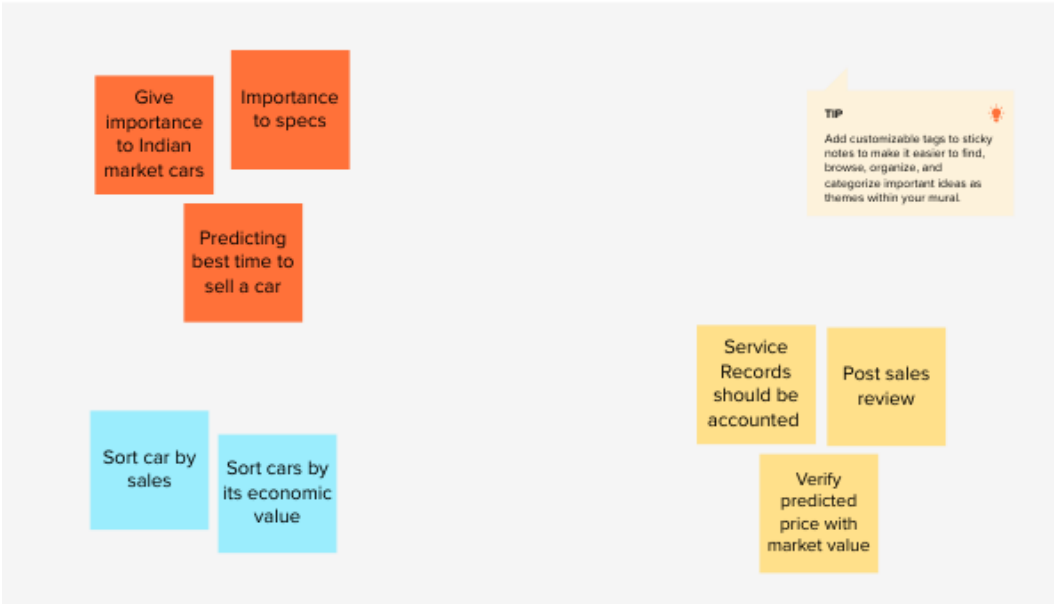
You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

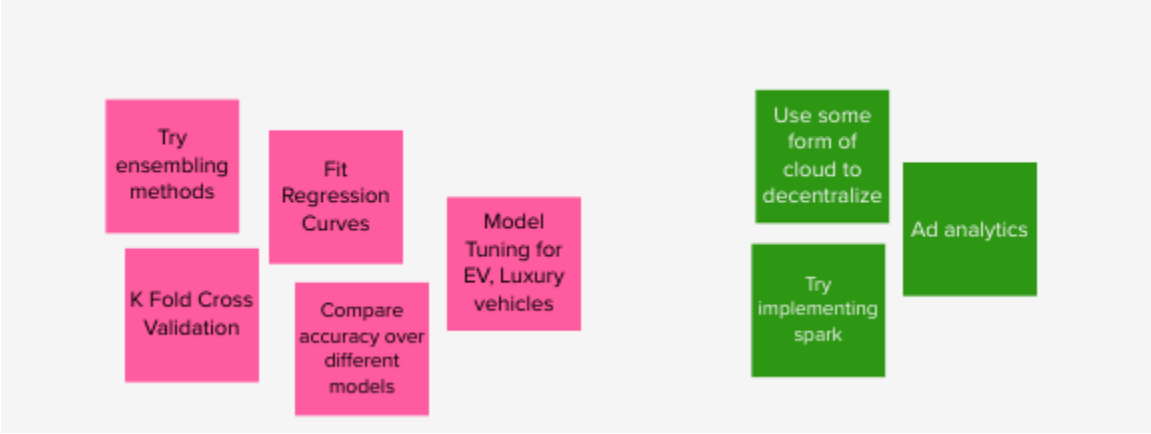
3

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





4

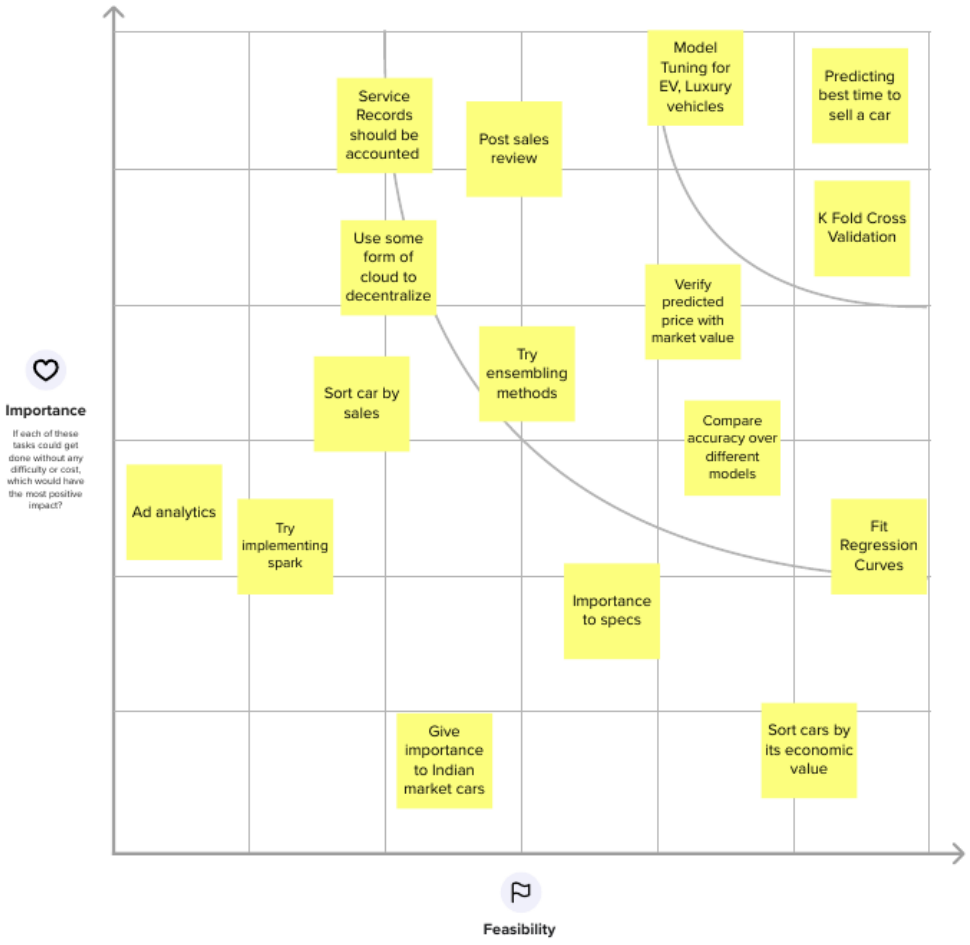
Prioritize

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

TIP

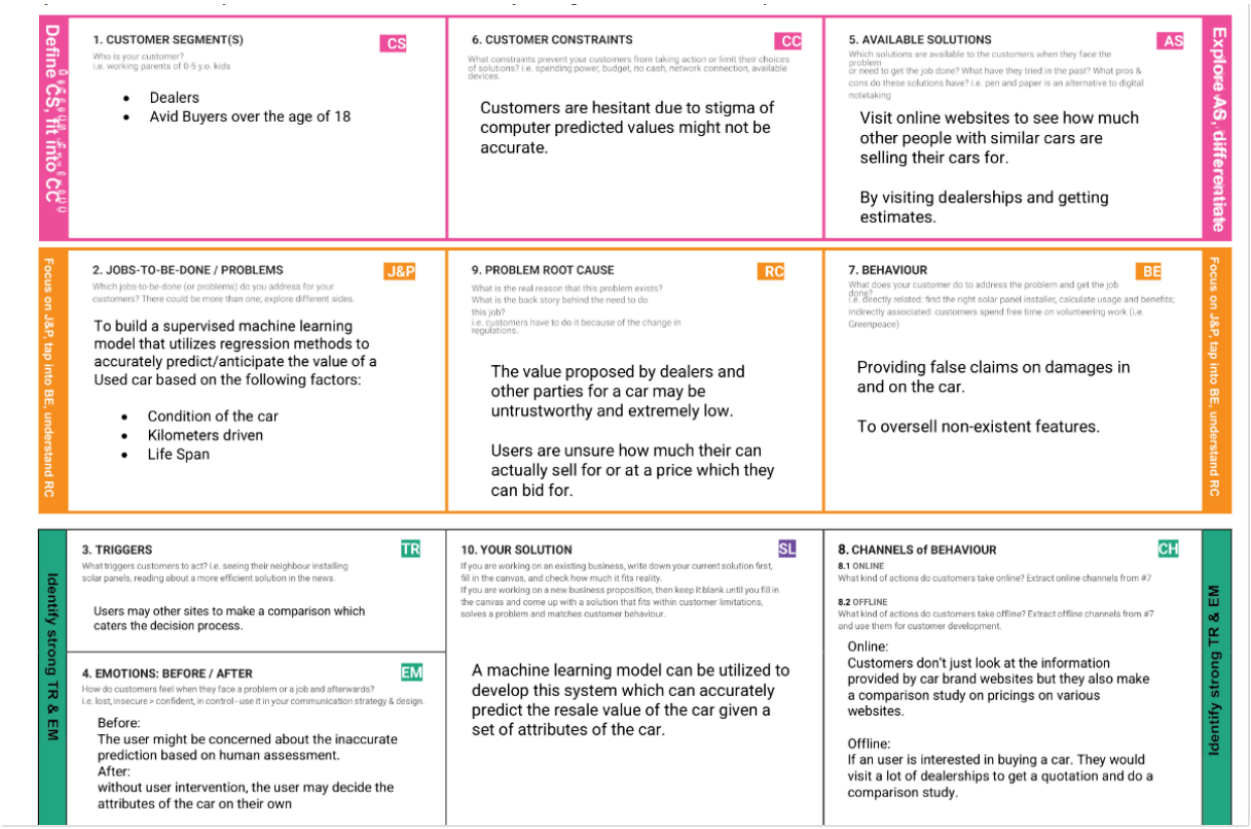
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.



3.3. Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	To predict the resale value of a second hand car or a used car considering its features.
2.	Idea / Solution description	To develop a Machine learning algorithm after comparative analysis, which predicts the best possible resale value of any used car which can be used through a web UI
3.	Novelty / Uniqueness	- The prediction engine contains a number of models. The more evaluation models that are used, the higher the performance. To further enhance performance, additional pre-processing processes are carried out. By allowing the dataset to train and test through all possible regression models, the model is finally finished. - Launching the project as a web app allows everyone to accurately assess their car's resale worth or for buyers to get an idea of the cost of the car they wish to buy
4.	Social Impact / Customer Satisfaction	- This acts as the solution for sellers and for buyers.. - The seller now has no need to get worried about the price that is required to resale. - The buyer could also accept the resale price of the car as the algorithm considers even minute factors. - They can rest easy as they know that there was no malpractice in setting the price. - Apart from this, one could also get to know about the features or attributes which are involved in detecting the resale value. - It decreases the workload of both seller and buyer.
5.	Business Model (Revenue Model)	This project can bring in revenue in 2 streams: - One through commission from the buyer as it helped facilitate the sale - The other is through ad analytics
6.	Scalability of the Solution	- Model could be hosted in cloud infrastructure to help handle higher load. - Depending on country slight changes can be made to features of the dataset to provide even better predictions - Distributed computing (Apache Spark)

3.4. Problem Solution fit



4. REQUIREMENT ANALYSIS

4.1. Functional requirements

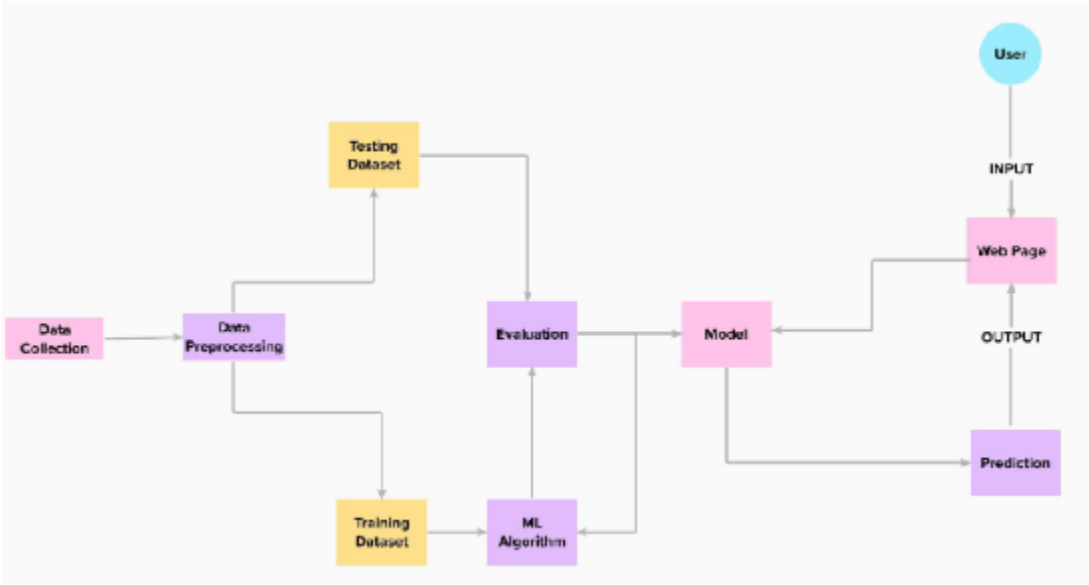
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail
FR-2	User Confirmation	Confirmation via Email
FR-3	User Details	Name, e-mail, address, preferences, etc
FR-4	Collect and Display Car Details	Fuel type, Manufactured year, Kms driven, Past Records
FR-5	Value Prediction	Car Resale Value Prediction

4.2. Non-Functional requirements

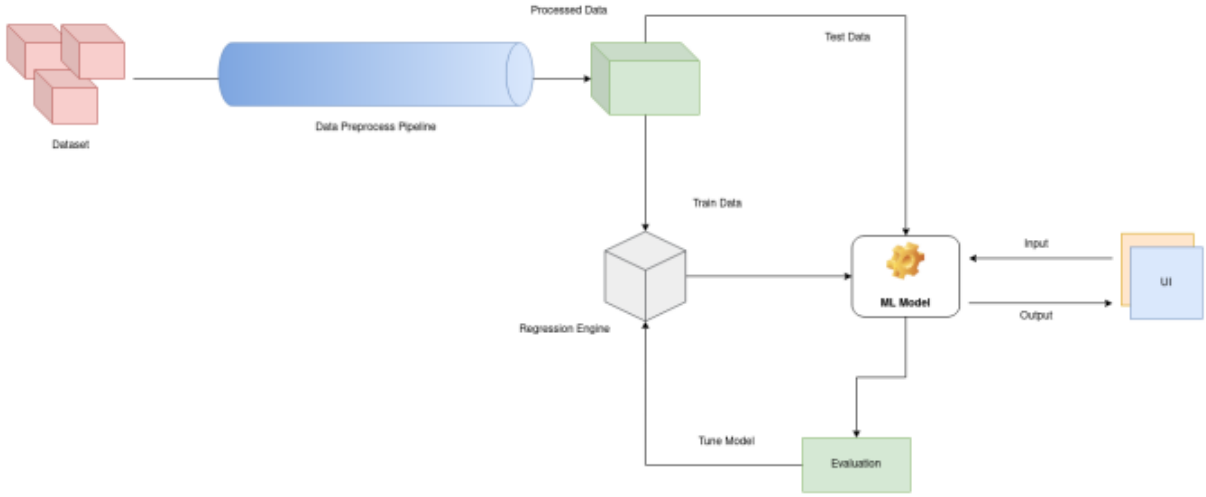
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Great UI(User Interface), Accuracy in value prediction
NFR-2	**Security**	Protect user password and Personal details
NFR-3	**Reliability**	Rate of occurrence of failure is less, Failure free.
NFR-4	**Performance**	Perform correct prediction value, The landing page support several users and must provide 5 seconds or less response time
NFR-5	**Availability**	Uninterrupted services must be available in all time except the time of server updation
NFR-6	**Scalability**	Can handle any amount of data and perform many computations in a cost-effective and timesaving way to instantly serve millions of users residing at global locations

5. PROJECT DESIGN

5.1. Data Flow Diagrams



5.2. Solution & Technical Architecture



5.3. User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Web user)	FR-1 User Registration	USN-1	As a user, I register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-3
Customer (Web user)	FR-4 Collect and Display Car Details	USN-2	As a user, I enter the car details in the application.	I can enter the car details in the application.	Medium	Sprint-3
Customer (Web user)	FR-5 Value Prediction	USN-3	As a user, I expect to see the predicted resale value.	I can see the predicted resale value.	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1. Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Dataset reading and Preprocessing	USN-2	Cleaning the dataset and splitting into dependent and independent variables.	1	High	4
Sprint-2	Build the model	USN-3	Use the appropriate model for building and saving the model(file format).	2	Low	4
Sprint-3	User Registration	USN-1	Users can register for the application by entering my email and confirming my password,phone number.	2	High	4
Sprint-3	Create and implement the application	USN-2	By using the flask framework to implement the model.	2	Medium	4
Sprint-4	Train the model with dataset in IBM cloud	USN-3	Finally train the model on IBM cloud and deploy the application	1	High	4
Sprint-4	Predict the price of used cars	USN-3	The application try to predict the price of used	2	High	4

6.2. Sprint Delivery Schedule

- Pre -requisites
- Import Required libraries
- Collect Data Set
- Pre the process the data
- Choose the Appropriate Model
- Train the model on IBM
- Integrate with Flask endpoint
- Index..html
- Registration form.html
- Flask application
- App.py

6.3. Reports from JIRA

CRV - Car Resale Value

CRVP Sprint 1

VP A

TO DO	IN PROGRESS	DONE 2 ISSUES
		Collect Dataset CRVP-1 A
		Pre Processing CRVP-2

CRVP Sprint 2

VP SB

TO DO	IN PROGRESS	DONE 1 ISSUE
		Choose Appropriate Model CRVP-3 SB

CRVP Sprint 3

VP A SB

TO DO	IN PROGRESS	DONE 3 ISSUES
		Flask App CRVP-4 VP
		Check Metrics CRVP-5 A
		Save Results CRVP-6 SB

CRVP Sprint 4

TO DO	IN PROGRESS	DONE 2 ISSUES ✓
		UI CRVP-7 ✓ SB
		UAT CRVP-8 ✓ VP

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

7.1. Model Selection

Several models were trained and the best was chosen.

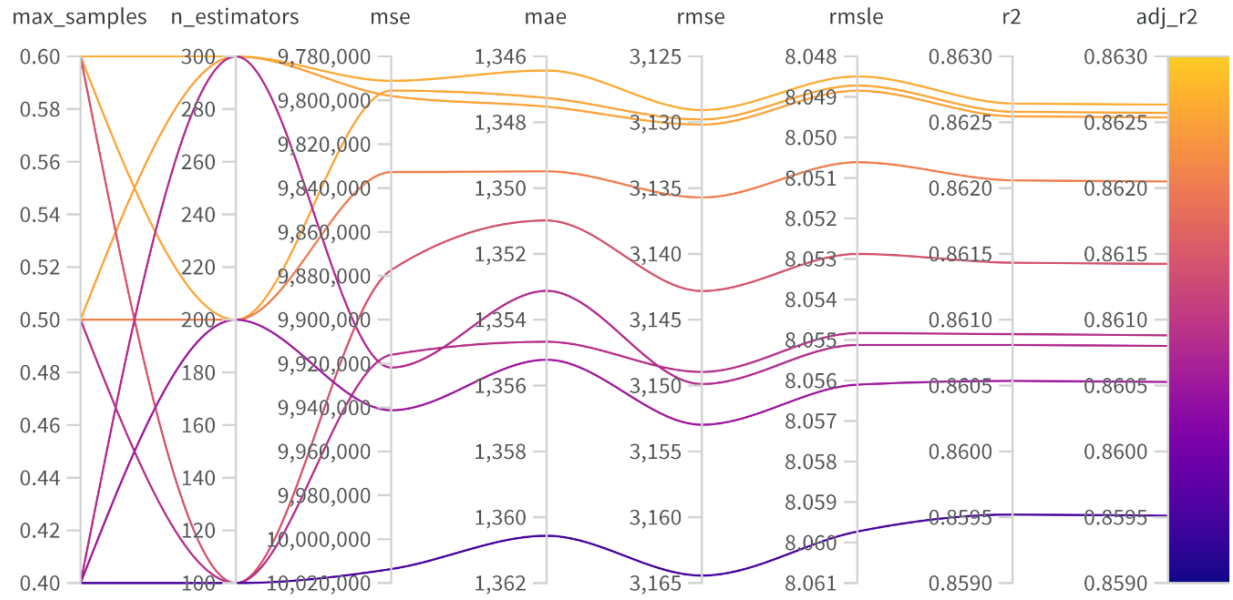
1. Bagging regressor
2. LGBM regressor
3. Random Forest regressor

Compared using WandB service as shown:

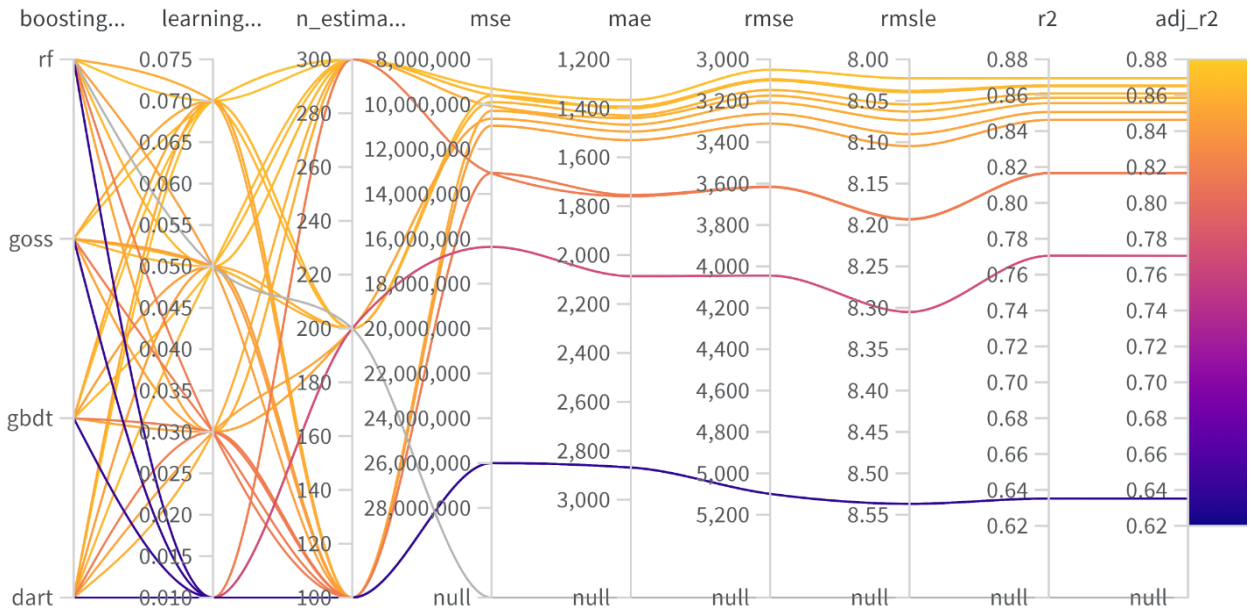
```
sweep_id = wandb.sweep(sweep=bagging_regressor_configs, project="car_resale_value")
wandb.agent(sweep_id=sweep_id, function=bagging_regressor)
```

These are the results:

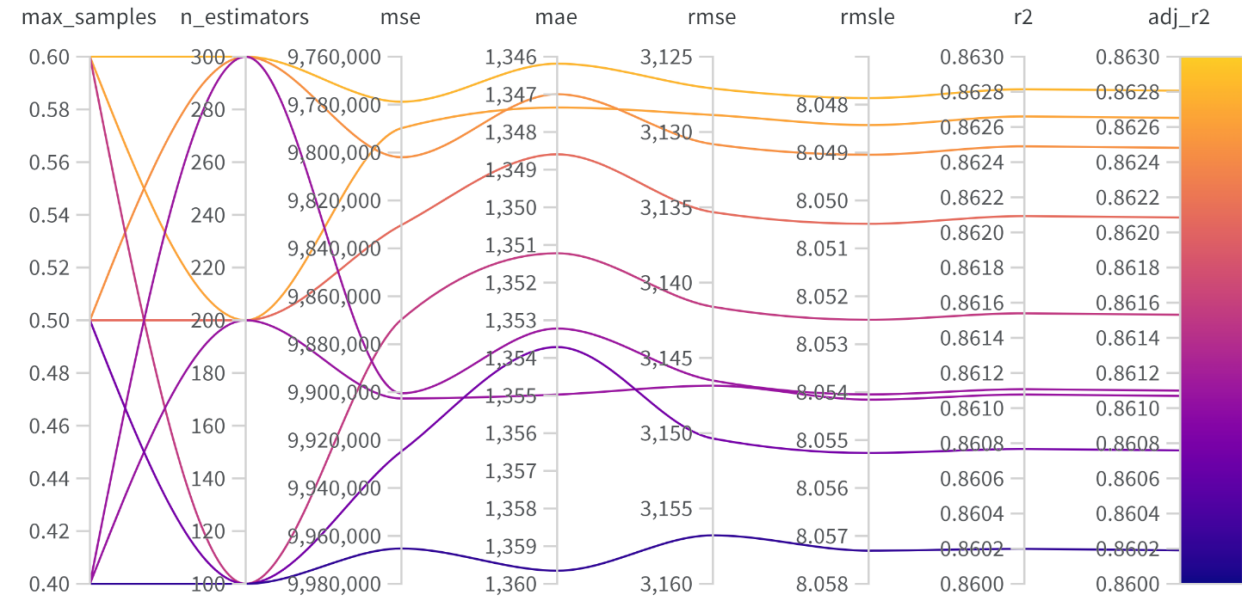
Bagging Regressor:



LGBM regressor:



Random Forest Regressor:



LGBM regressor was chosen

7.2. Model Training in IBM and deployment

Model was then trained on IBM watson and was exposed using an API key.

1. Create IBM account
2. Create new deployment
3. Upload assets
4. Run jupyter notebook
5. Deploy model API

```
deployment_props = {
    client.deployments.ConfigurationMetaNames.NAME:DEPLOYMENT_NAME,
    client.deployments.ConfigurationMetaNames.ONLINE: {}
}
```

```
deployment = client.deployments.create(
    artifact_uid=model_id,
    meta_props=deployment_props
)
```

```
#######################################################################################

Synchronous deployment creation for uid: '2db06ed2-1c6e-4faf-8163-843ed9699bfb' started

#######################################################################################


initializing
Note: online_url is deprecated and will be removed in a future release. Use serving_urls instead.

ready


------------------------------------------------------------------------------------------------
Successfully finished deployment creation, deployment_uid='660695dd-d0d8-419f-8850-76ccf4172103'
------------------------------------------------------------------------------------------------
```

7.3 Integrate API and Flask app

Then said API was integrate with flask app.

```
def APIConnect(apikey):
    API_KEY = apikey
    token_response = requests.post('https://iam.cloud.ibm.com/identity/token', data={"apikey":API_KEY, "grant_type":
    mltoken = token_response.json()["access_token"]
    header = {'Content-Type': 'application/json', 'Authorization': 'Bearer ' + mltoken}

    return API_KEY, token_response, mltoken, header

API_KEY, token_response, mltoken, header = APIConnect("8MQKoaCV7PXi-bUaQppxftdf1gASUWX9QOSHs-dBSWkJ")
```

7.4 Home page

Home page provide a simple User login

```
<body>
  <section class="header">
    <div class="form">
      <div class="title">Car resale value Predictor</div>
      <div class="subtitle">Best system to predict the amount of resale value based on the parameters provided by the
        user.</div>
      <form name="myForm" action="/predict_page" method="POST">
        <div class="input-container ic1">
          <input id="username" class="input" type="text" placeholder=" "/>
          <div class="cut"></div>
          <label for="firstname" class="placeholder">Username</label>
        </div>
        <div class="input-container ic2">
          <input id="pwd" class="input" type="password" placeholder=" "/>
          <div class="cut"></div>
          <label for="firstname" class="placeholder">Password</label>
        </div>
        <center>
          <input id="submit" class="submit" type="button" value="Login & Check Price" onclick="validate()"/>
        </center>
    </div>
```

```
<script>
  var attempt = 5; // Variable to count number of attempts.
  // Below function Executes on click of login button.
  function validate() {
    var username = document.getElementById("username").value;
    var password = document.getElementById("pwd").value;
    if (username == "admin" && password == "admin@123") {
      alert("Login successfully");
      window.location = "/predict_page"; // Redirecting to other page.
      return false;
    }
    else {
      attempt--;// Decrementing by one.
      alert("Invalid Credentials!\n" + attempt + " attempts left");
      // Disabling fields after 5 attempts.
      if (attempt == 0) {
        document.getElementById("username").disabled = true;
        document.getElementById("pwd").disabled = true;
        document.getElementById("submit").disabled = true;
        return false;
      }
    }
  }
</script>
```

7.4 Receive Car Details

Next the user is prompted to enter car details.

```
<body>
    <div class="form">
      <div class="ti">Let's Get a Accurate Resale Value of Your Car</div>
      <div class="su">r Car details</div>
      <form action="http://localhost:5000/predict" method="GET">
      <div class="input-container ic1">
        <input id="firstname" class="input" type="text" placeholder=" " name="regyear"/>
        <div class="cut"></div>
        <label for="firstname" class="placeholder">Registration year</label>
      </div>
      <div class="input-container ic2">
        <input id="firstname" class="input" type="text" placeholder=" " name="regmonth"/>
        <div class="cut"></div>
        <label for="firstname" class="placeholder">Registration month</label>
      </div>
      <div class="input-container ic2">
        <input id="firstname" class="input" type="text" placeholder=" " name="powerps"/>
        <div class="cut"></div>
        <label for="firstname" class="placeholder">Power (PS)</label>
      </div>
      <div class="input-container ic2">
        <input id="lastname" class="input" type="text" placeholder=" " name="kms"/>
        <div class="cut"></div>
        <label for="lastname" class="placeholder">KMs Driven</label>
      </div>
      <div class="ic3"style="font-family: sans-serif; font-size:18px; padding-left:20px">
        <label for="lastname">Transmission Type</label><br>
        <input type="radio" name="geartype" value="manual"/> Manual
```

Follow link (ctrl + click)

These inputs are then read in app.py to then be pre-processed and given to model API

```python
def getInputs(request):
    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')

    return reg_year, reg_month, powerps, kms, gearbox, damage, model, brand, fuel_type, veh_type
```

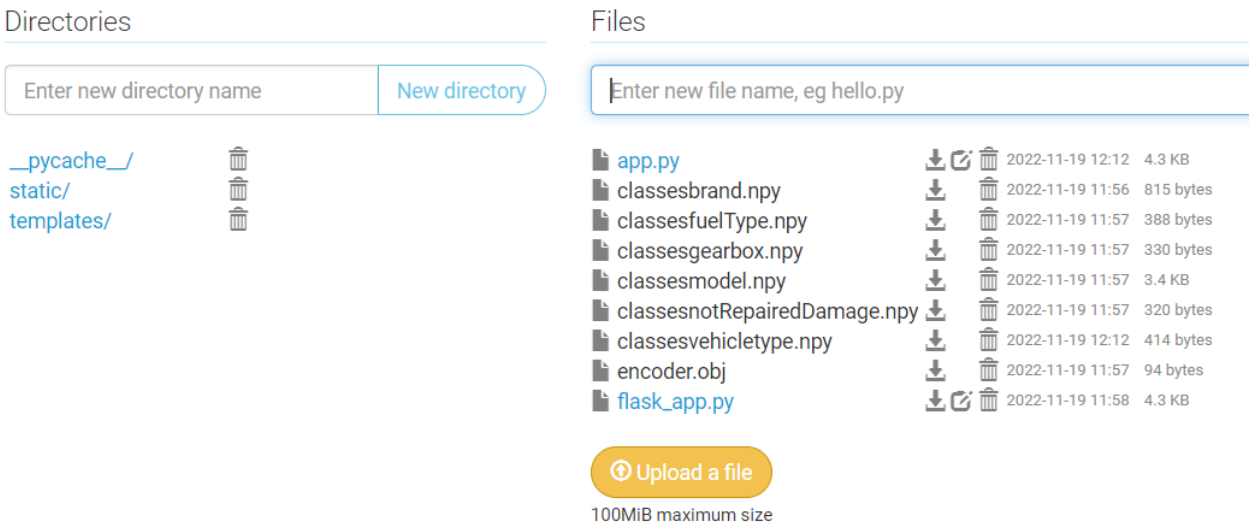
```python
# manually define and pass the array(s) of values to be scored in the next line
payload_scoring = {"input_data": [{"fields": [['yearOfReg', 'powerPS', 'kilometer', 'monthO

# send inout data to model API end point
response_scoring = requests.post('https://jp-tok.ml.cloud.ibm.com/ml/v4/deployments/660695c
#retrieve response
predictions = response_scoring.json()
print(response_scoring.json())
predict = predictions['predictions'][0]['values'][0][0]
#convert to INR
predict*=84.82
predict=round(float(predict),2)

# send relevant variables to predit.html
return render_template('predict.html',**locals())
```

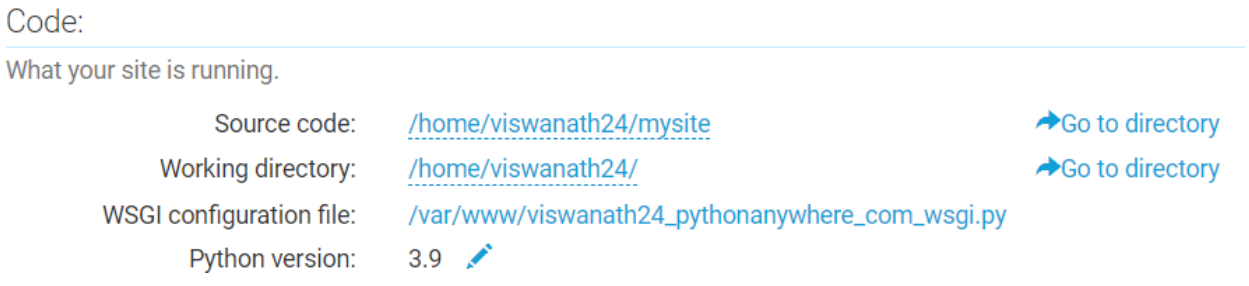
7.5 Result from model

The result from model is then given to final page for predicted price

```
<body>
  <section class="header">
      <nav>
      </nav>
        <div class="text-box">
<center>
      <table id="value_table">
        <tr>
          <th>Registration</th>
          <td>{{reg_month}} {{reg_year}}</td>
        </tr>
        <tr>
          <th>Power</th>
          <td>{{powerps}}</td>
        </tr>
        <tr>
          <th>Kilometers driven</th>
          <td>{{kms}}</td>
        </tr>
        <tr>
          <th>Gear Type</th>
          <td>{{gearbox}}</td>
        </tr>
        <tr>
          <th>Repaired/Damaged</th>
          <td>{{damage}}</td>
        </tr>
        <tr>
```

7.6 Host Application





8. TESTING

8.1. Test Cases

				Date	3-Nov-22								
				Team ID	PNT2022TMID35625								
				Project Name	Project - Car Resale Value Prediction								
				Maximum Marks	4 marks								
Test case ID	**Feature Type**	**Component**	**Test Scenario**	**Pre-Requisite**	**Steps To Execute**	**Test Data**	**Expected Result**	**Actual Result**	**Status**	**Comments**	**TC for Automation(Y/N)**	**BUG ID**	**Executed By**
LoginPage_TC_OO1	Functional	Home Page	Verify user is able to see the Login popup when launching the web-app		1.Enter URL and click go 2.Verify login/Singup popup displayed or not	https://viswanath24.pythonanywhere.com	Login Form is displayed	Working as expected	Pass				Sudharsan B, Viswanath Pathmanaban
LoginPage_TC_OO2	Non-functional	Home Page	Verify the UI elements in Login are readable and designed well		1.Enter URL and click go 2.Verify login popup with below UI elements: a. Username b. password c. login button	https://viswanath24.pythonanywhere.com	Application should show below UI elements: a.Username text box b.password text box c.Login button with blue colour	Working as expected	Pass	Are elements designed well?			Shenbaga Shudhan, Rukesh A
LoginPage_TC_OO3	Functional	Home page	Verify user is able to log into application with Valid credentials		1.Enter URL click go 2.Enter InValid username in Username text box 3.Enter valid password in password text box 4.Click on login button	Username: admin password: admin@123	User should navigate to the car details page	Working as expected	Pass				Sudharsan B, Viswanath Pathnaban
LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter URL click go 2.Enter InValid username in Username text box 3.Enter valid password in password text box 4.Click on login button	Username: soosanexe password: admin@123	Application should show alert for INVALID credentials	working as expected	Pass				Shenbaga Shudhan, Sudharsan B
LoginPage_TC_OO4	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter and click go 2.Enter Valid username/email in Email text box 3.Enter Invalid password in password text box 4.Click on login button	Username: admin password: Testing1236786867868768 76	Application should show alert for INVALID credentials	working as expected	Pass				viswanath, rukesh
LoginPage_TC_OO5	Functional	Login page	Verify user is able to log into application with InValid credentials		1.Enter and click go 2.Enter Invalid username/email in Email text box 3.Enter Invalid password in password text box 4.Click on login button	Username: soosanexe password: admin@12345	Application should show alert for INVALID credentials	working as expected	Pass				Shenbaga Shudhan, Rukesh A
DetailsPage_TC_001	Functional	Details Page	Verify User is able to see UI elements		1. Login from login page	https://viswanath24.pythonanywhere.com	User should be able to see the UI elements render	working as expected	Fail	~~CSS issues.~~ **FIXED**		BUG_01	Sudharsan B, Viswanath Pathmanaban
DetailsPage_TC_002	Functional	Details Page	User is able to Enter Details into input fields		1. Login from login page	https://viswanath24.pythonanywhere.com	User should be able to enter Data into input fields	working as expected	Pass				Shenbaga Shudhan, Rukesh A
DetailsPage_TC_003	Non-functional	Details Page	Verify the UI elements are readable and designed well		1. Login from login page	https://viswanath24.pythonanywhere.com	Application should show below UI elements: a. Reg year box b.Reg month box etc	unexpected	Fail	~~CSS bugs.~~ **FIXED**		BUG_02	Sudharsan B, Viswanath Pathmanaban, Shenbaga Shudhan, Rukesh A
PredictPage_TC_001	Functional	Predict Page	User is able to UI elements		1. Login from login page 2. Enter car details 3. Press submit	https://viswanath24.pythonanywhere.com	User is able to navigate into Final Page	unexpected	Fail	~~Page is redirected to localhost.~~ **FIXED**		BUG_03	Sudharsan B, Viswanath Pathmanaban, Shenbaga Shudhan, Rukesh A
		Predict Page	User able to see mode prediction		1. Login from login page 2. Enter car details 3. Press submit	https://viswanath24.pythonanywhere.com	User is able to see table of parameters and predicted price	working as expected	Pass				Sudharsan B, Viswanath Pathmanaban

8.2. User Acceptance Testing

1. Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	3	0	0	2	5
Duplicate	1	0	3	0	4
External	8	3	0	1	12
Fixed	12	3	3	3	21
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	0	2	1	3
Totals	25	6	10	8	77

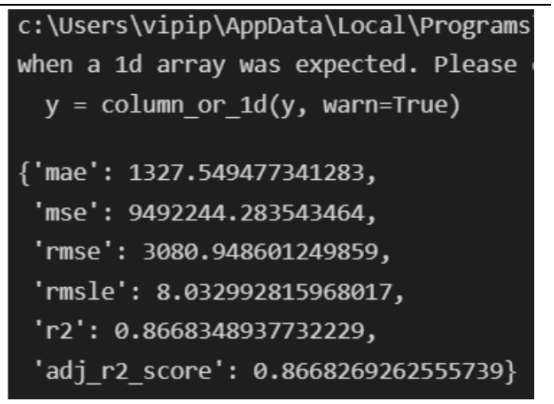
2. Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Login Page	7	0	0	7
Details Page	3 major (60+ total)	20	0	3 + 40
Predict Page	3 major (60+ total)	20	0	3 + 40
Security	3	0	0	3

9. RESULTS

9.1. Performance Metrics

S.No.	Parameter	Values	Screenshot
1.	Metrics	**Regression Model:** MAE - 1327.5, MSE - 9492244, RMSE - 3080, R2 score - 0.86	

10. ADVANTAGES & DISADVANTAGES

Advantages

- Several models were trained and best was chosen
- Predicted price was close to actual pricing
- Model is hosted in IBM, so rest assured that it will be performant

CONS:

- Only one user available as of now

11. CONCLUSION

Due to rising new car prices and buyers' financial inability to purchase them, used car sales are on the rise globally. As a result, there is an urgent need for a Used Car Price Prediction system that accurately analyzes the worthiness of the vehicle based on a range of factors. The suggested approach will aid in determining the most accurate used automobile price estimate.

12. FUTURE SCOPE

In future this machine learning model may bind with various websites 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 a user interface for interacting with users. 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.

13. APPENDIX

Source Code

GitHub & Project Demo Link