Car Resale Value Prediction

Applied Data Science

Submitted by

Sidharth G

Abijeeth Vasra T R

Arun Kalyan M

Kanishmithran J

Dinesh R R

Project ID PNT2022MID21557

Department of Information Technology,
Thiagarajar College of Engineering, Madurai - 625014.

Institution Mentor: Dr. C. Jeyamala
Industrial Mentor: Prof Swetha

1.Introduction:

1.1 Project Overview:

The ability to anticipate a car's resale value based on characteristics such as miles driven and fuel type is helpful to users.

The sole goal of this general-purpose system for estimating resale value is to estimate the amount that the user can probably acquire.

So that the user can get an estimated value before reselling the car and avoid making a deal at a loss, we attempt to predict the amount of resale with the highest degree of accuracy.

Project Flow:

1. To enter the input features, the user engages with the user interface (UI).
2. The integrated model analyzes the features of the entered input.
3. The prediction is displayed on the UI when the model has processed the input.

1.2 Purpose:

Making a system to forecast car resale value is primarily intended as a way to practice Python using Data Science. The system that forecasts the amount of resale value for cars is based on the user-provided parameters. The car's details are entered into the provided form by the user, and the value at which it will be sold is then predicted.

2.Literature Survey

2.1 Existing Problem

1) Predicting the Price of Used Cars using Machine Learning
2) Used Cars Price Prediction using Supervised Learning Techniques
3) Car Price Prediction Using Machine Learning
4) Used Car Price Prediction using K-Nearest Neighbor Based Model

2.2 References

http://ripublication.com/irph/ijict_spl/ijictv4n7spl_17.pdf

https://www.researchgate.net/publication/343878698_Used_Cars_Price_Prediction_using_Supervised_Learning_Techniques

https://www.jetir.org/view?paper=JETIR2204621

https://www.ijirase.com/assets/paper/issue_1/volume_4/V4-Issue-3-686-689.pdf

2.3 Problem Statement Definition

1. This project deals with the application of supervised machine learning techniques to predict the price of used cars in Mauritius.

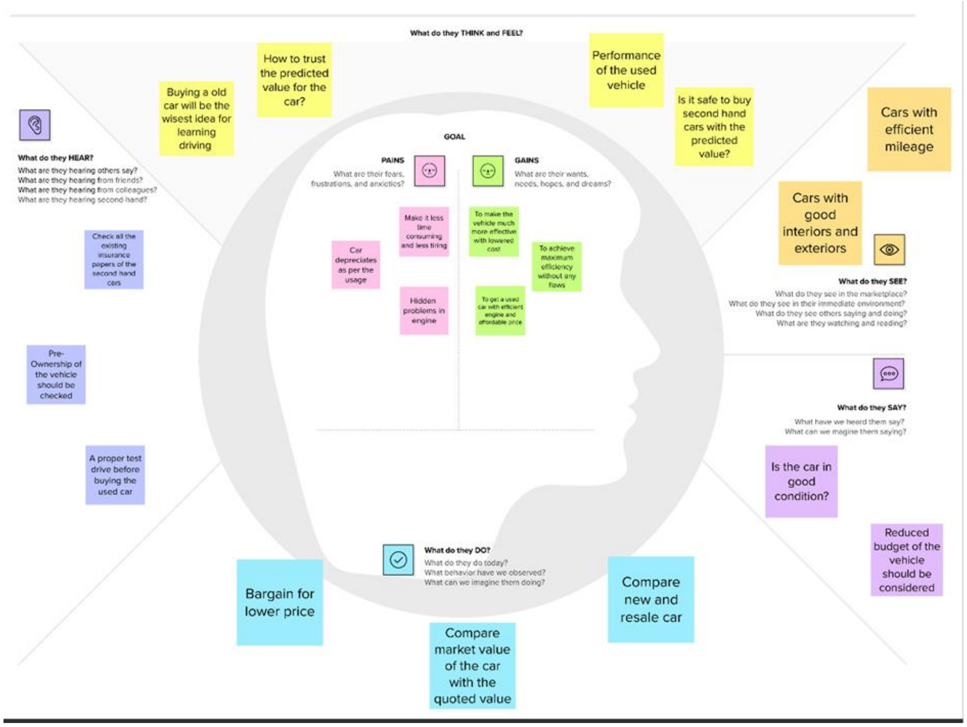
2. Using Multiple Regression and Regression trees, the project tries to develop a statistical model which will be able to predict the price of a used car, based on previous consumer data and a given set of features. This project will also be comparing the prediction accuracy of these models to determine the optimal one

3. This project is going to predict the car cost with the help of machine learning algorithms which are made available by a python environment such as the Gradient Boosting Algorithm. The dataset comprises data related to different car brands with a set of parameters. The primary purpose is to design a model for a given dataset and predict the car price with better accuracy.

4. This paper proposed a supervised machine learning model using KNN (K Nearest Neighbor) regression algorithm to analyze the price of used cars. Through this experiment, the data was examined with different trained and test ratios. As a result, the accuracy of the proposed model is around 85%

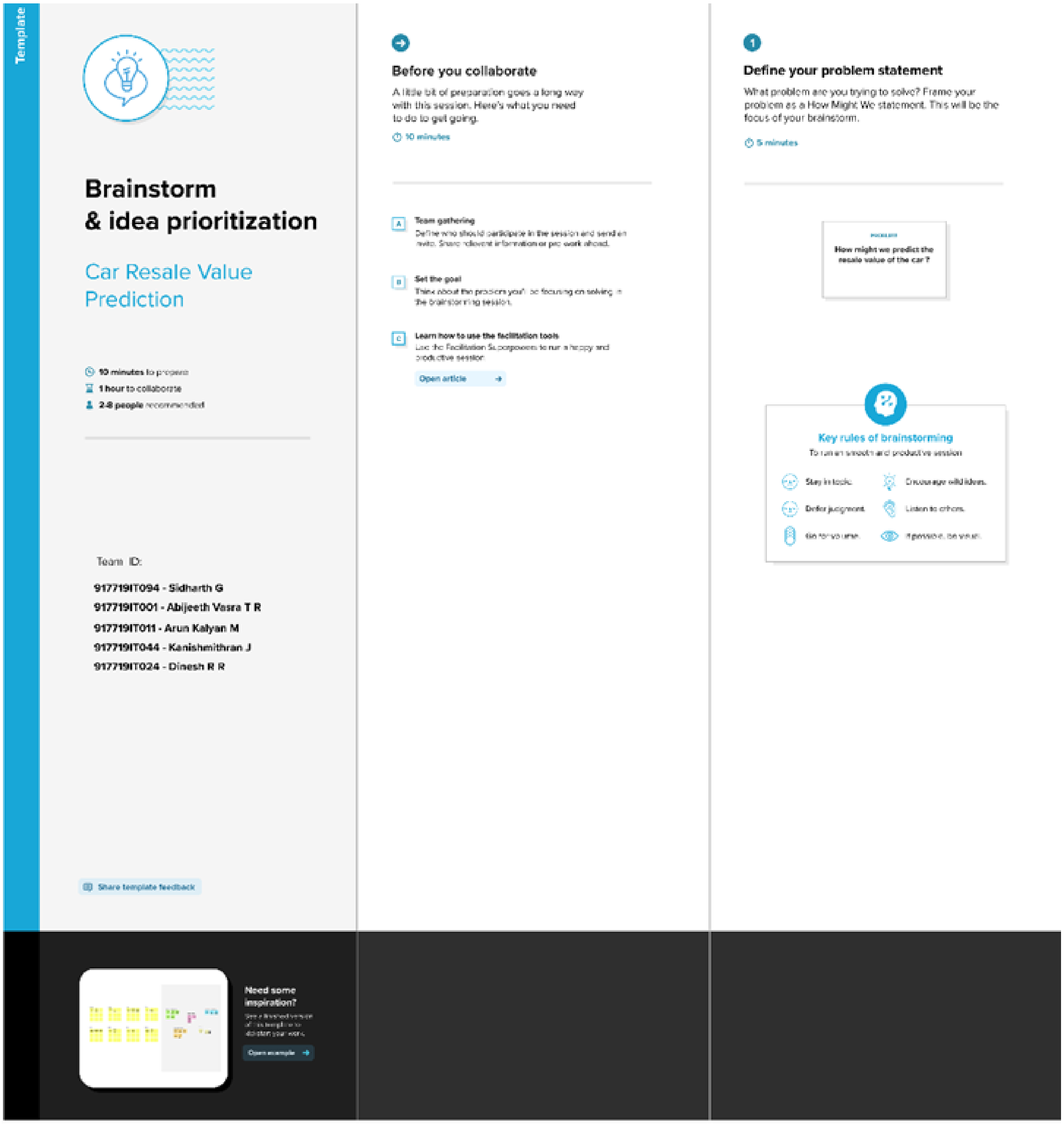
3.Ideation and Proposed Solution:

3.1 Empathy Map Canvas


What do they THINK and FEEL?
Buying a old car will be the wisest idea for learning driving
How to trust the predicted value for the car?
Performance of the used vehicle
Is it safe to buy second hand cars with the predicted value?
Cars with efficient mileage
What do they HEAR?
What are they hearing others say?
What are they hearing from friends?
What are they hearing from colleagues?
What are they hearing second-hand?
Check all the existing insurance papers of the second hand cars
Pre-Ownership of the vehicle should be checked
A proper test drive before buying the used car
GOAL
PAINS
What are their fears, frustrations, and anxieties?
Make it less time consuming and less tiring
Car depreciates as per the usage
Hidden problems in engine
GAINS
What are their wants, needs, hopes, and dreams?
To make the vehicle much more effective with lowered cost
To achieve maximum efficiency without any flaws
To get a used car with efficient engine and affordable price
Cars with good interiors and exteriors
What do they SEE?
What do they see in the marketplace?
What do they see in their immediate environment?
What do they see others saying and doing?
What are they watching and reading?
What do they SAY?
What have we heard them say?
What can we imagine them saying?
Is the car in good condition?
Reduced budget of the vehicle should be considered
What do they DO?
What do they do today?
What behavior have we observed?
What can we imagine them doing?
Bargain for lower price
Compare market value of the car with the quoted value
Compare new and resale car

3.2 Ideation and Brainstorming

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



Template

Brainstorm
& idea prioritization

Car Resale Value
Prediction

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Team ID:

917719IT094 - Sidharth G
917719IT001 - Abijeeth Vasra T R
917719IT011 - Arun Kalyan M
917719IT044 - Kanishmithran J
917719IT024 - Dinesh R R

Share template feedback

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1 **Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM
How might we predict the resale value of the car ?

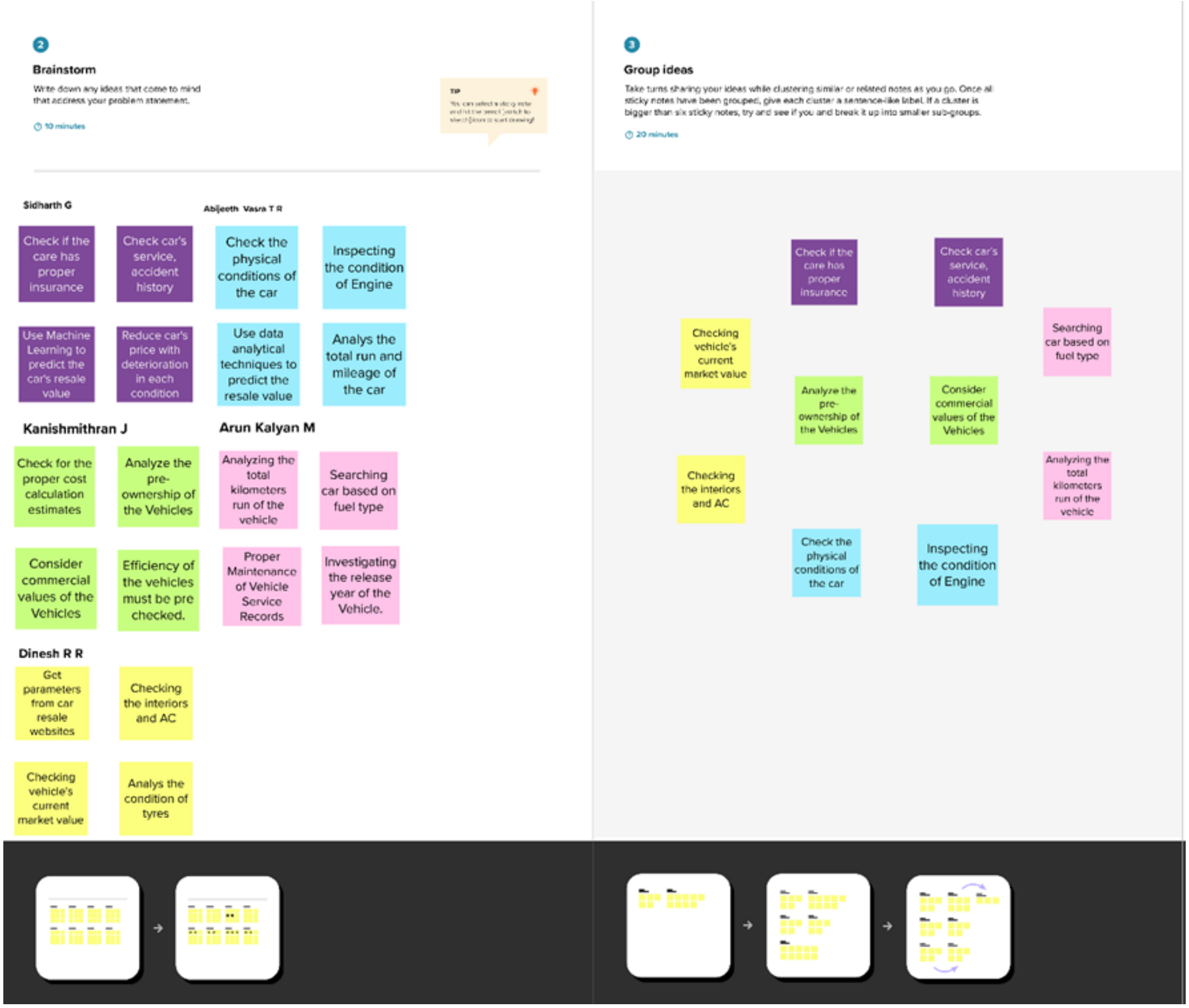
Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

Need some inspiration?
See a finished version of this template to kickstart your work.

Open example →

Step-2: Brainstorm, Idea Listing and Grouping



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!

Sidharth G

- Check if the care has proper insurance
- Check car's service, accident history
- Use Machine Learning to predict the car's resale value
- Reduce car's price with deterioration in each condition

Abijeeth Vasra T R

- Check the physical conditions of the car
- Inspecting the condition of Engine
- Use data analytical techniques to predict the resale value
- Analys the total run and mileage of the car

Kanishmithran J

- Check for the proper cost calculation estimates
- Analyze the pre-ownership of the Vehicles
- Consider commercial values of the Vehicles
- Efficiency of the vehicles must be pre checked.

Arun Kalyan M

- Analyzing the total kilometers run of the vehicle
- Searching car based on fuel type
- Proper Maintenance of Vehicle Service Records
- Investigating the release year of the Vehicle.

Dinesh R R

- Get parameters from car resale websites
- Checking the interiors and AC
- Checking vehicle's current market value
- Analys the condition of tyres

3

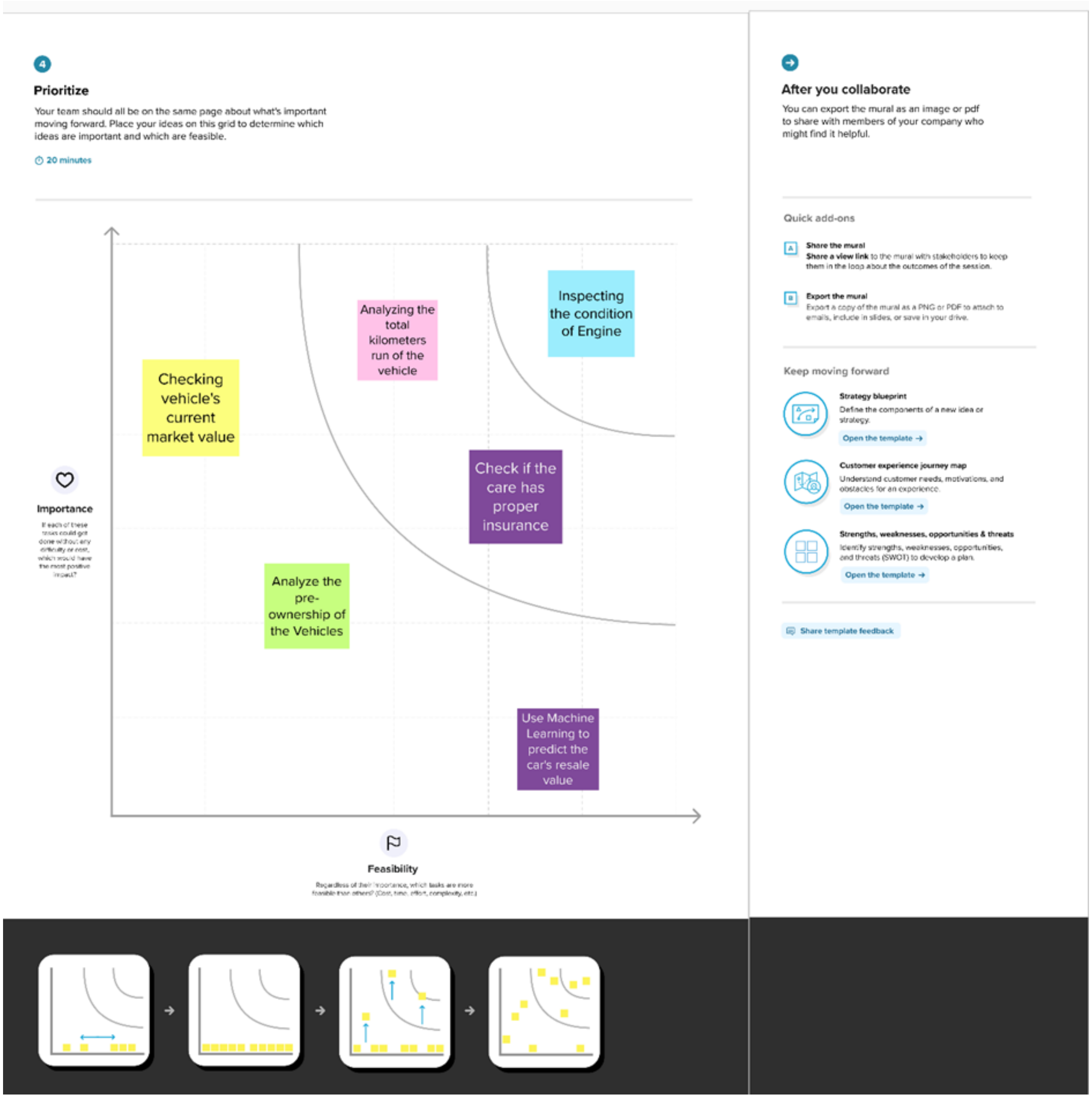
Group ideas

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

- Check if the care has proper insurance
- Check car's service, accident history
- Checking vehicle's current market value
- Searching car based on fuel type
- Analyze the pre-ownership of the Vehicles
- Consider commercial values of the Vehicles
- Checking the interiors and AC
- Analyzing the total kilometers run of the vehicle
- Check the physical conditions of the car
- Inspecting the condition of Engine

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

Analyzing the total kilometers run of the vehicle
Inspecting the condition of Engine
Checking vehicle's current market value
Check if the care has proper insurance
Analyze the pre-ownership of the Vehicles
Use Machine Learning to predict the car's resale value
Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?
Feasibility
Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

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

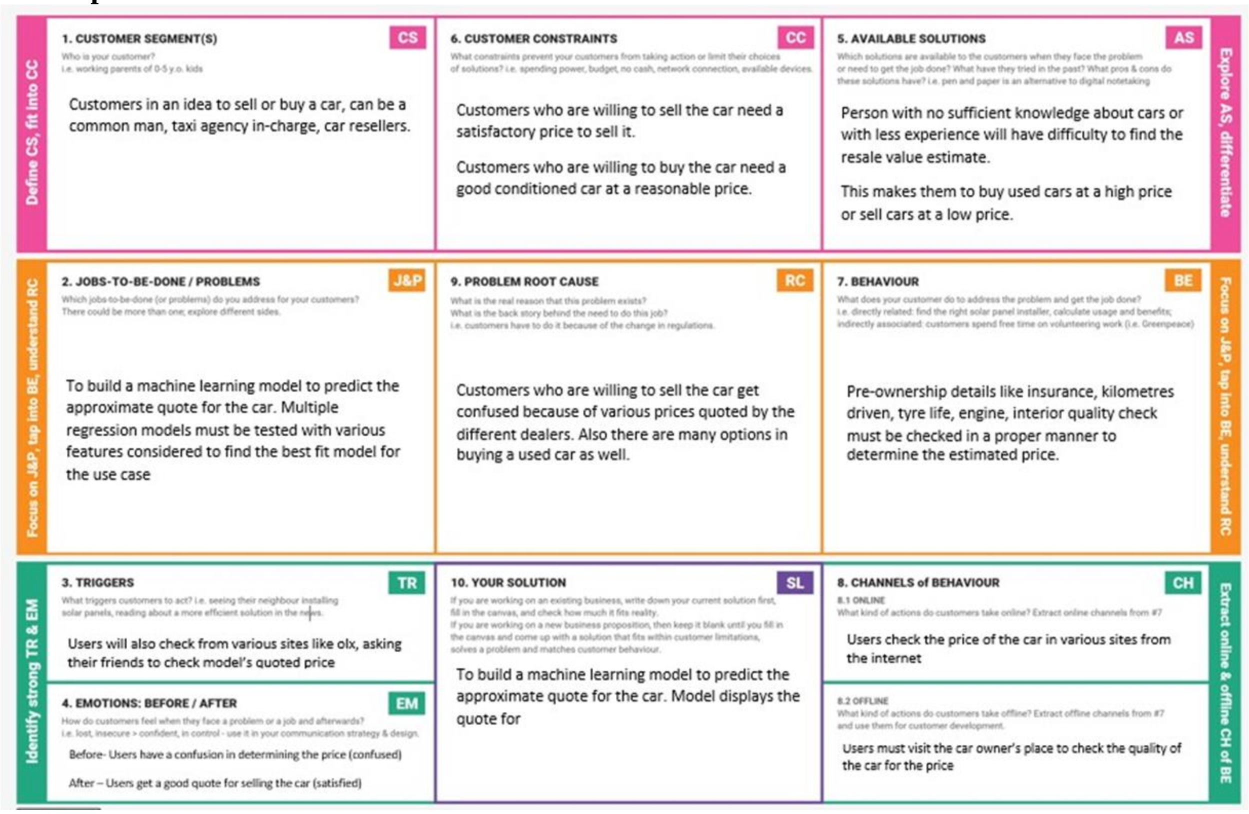
Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

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



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

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

Customers in an idea to sell or buy a car, can be a common man, taxi agency in-charge, car resellers.

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.

Customers who are willing to sell the car need a satisfactory price to sell it.

Customers who are willing to buy the car need a good conditioned car at a reasonable price.

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

Person with no sufficient knowledge about cars or with less experience will have difficulty to find the resale value estimate.

This makes them to buy used cars at a high price or sell cars at a low price.

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 machine learning model to predict the approximate quote for the car. Multiple regression models must be tested with various features considered to find the best fit model for the use case

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.

Customers who are willing to sell the car get confused because of various prices quoted by the different dealers. Also there are many options in buying a used car as well.

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)

Pre-ownership details like insurance, kilometres driven, tyre life, engine, interior quality check must be checked in a proper manner to determine the estimated price.

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 will also check from various sites like olx, asking their friends to check model's quoted price

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- Users have a confusion in determining the price (confused)

After – Users get a good quote for selling the car (satisfied)

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.

To build a machine learning model to predict the approximate quote for the car. Model displays the quote for

8. CHANNELS of BEHAVIOUR CH

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

Users check the price of the car in various sites from the internet

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

Users must visit the car owner's place to check the quality of the car for the price

Extract online & offline CH of BE

3.4 Problem Solution Fit

S. No	Parameter	Description
1.	Problem Statement (Problem to be solved)	There are a lot of ambiguities in predicting the precise resale value of the car. Since it involves only manual calculation with human assumed factors.
2.	Idea / Solution description	The way a machine predicts a value based on certain factors is way better than arbitrary manual prediction, thus machine learning can be used to predict the value of a resalable car by including certain factors as attributes(Dependent variables) to the model.

3.	Novelty / Uniqueness	The factors included are cumulative in nature, various factors affecting a car value is grouped into a single factor based on similarity (Insurance documents, RC book maintenance are grouped to a factor credibility) those factors are used for prediction
4.	Social Impact / Customer Satisfaction	Enables the customer to get a good idea about the used car prices and the features offered hence it motivates them to buy it.
5.	Business Model (Revenue Model)	Almost 5 million used cars are being sold every year in india, thus the market of car resales is huge and thousands of car resale dealers who are in need of such digital and efficient solution to predict the resale value of a car
6.	Scalability of the Solution	Since it involves generalized attributes and not attributes associated with a particular type of car model or car manufacturer, this can be applied to predict the resale value of any car, even any vehicle with few limitations.

4. Requirement Analysis

4.1 Function Requirement

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 Form Registration through Gmail Registration through Linkedin

FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Profile	View User’s Personal details Add a car to their favorites list
FR-4	Car Registration	User can input information like car’s date of purchase, price, damages incurred etc.
FR - 5	Viewing Past Predictions	Users are able to view past predictions for the price of the car. (Graph displaying the price of the car for a month)

4.2 Non-function Requirement

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

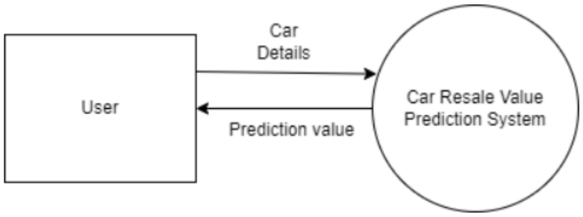
FR No.	**Non-Functional Requirement**	**Description**
NFR-1	**Usability**	A Simple and effective User Interface with proper layout and good usage of icons ensures each user finds it easy to access and interact with the system.
NFR-2	**Security**	Ensures all the user credentials should be protected and there should be a mandatory password strength check while creating password. Two factor Authentication methods can also be used.

NFR-3	**Reliability**	The ML model which is responsible for predicting the price of the car should be reliable. Model should be accurate enough to predict prices. Error rate should be as minimum as possible.
NFR-4	**Performance**	The system must provide a web page rendering images and texts upon receiving a request within a time of 8 seconds over a standard internet connection.
NFR-5	**Availability**	The website should be available to users 24x7. Any issues or errors will be addressed within the next 24 hours.
NFR-6	**Scalability**	The system must be scalable enough to support 1,00,000 requests at the same time without crashing.

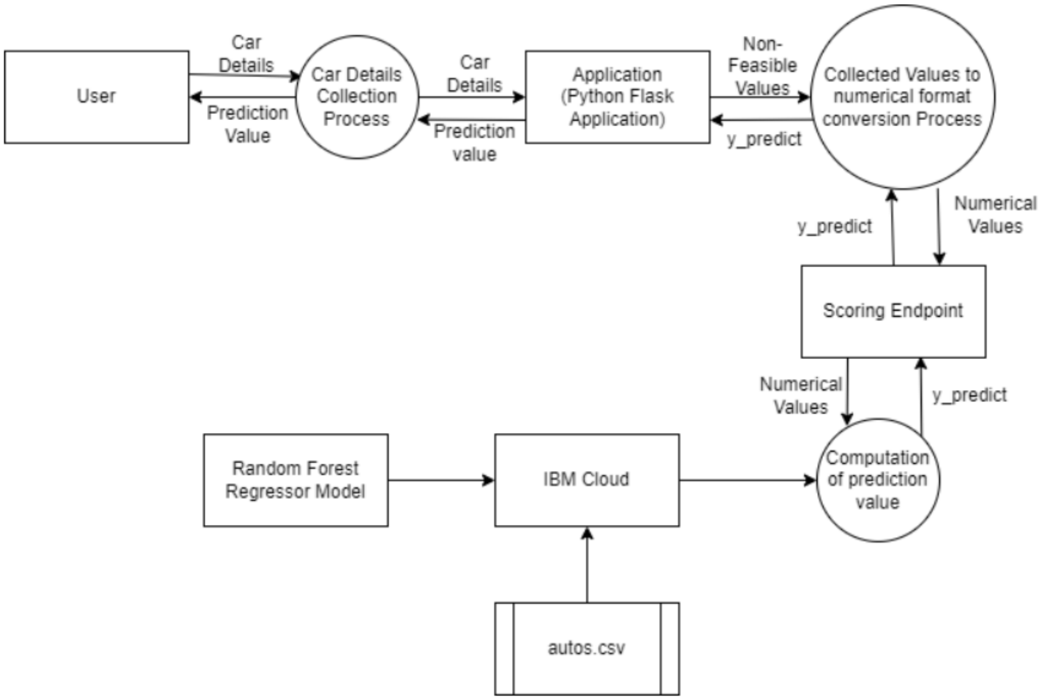
5.Project Design

5.1 Data Flow diagrams

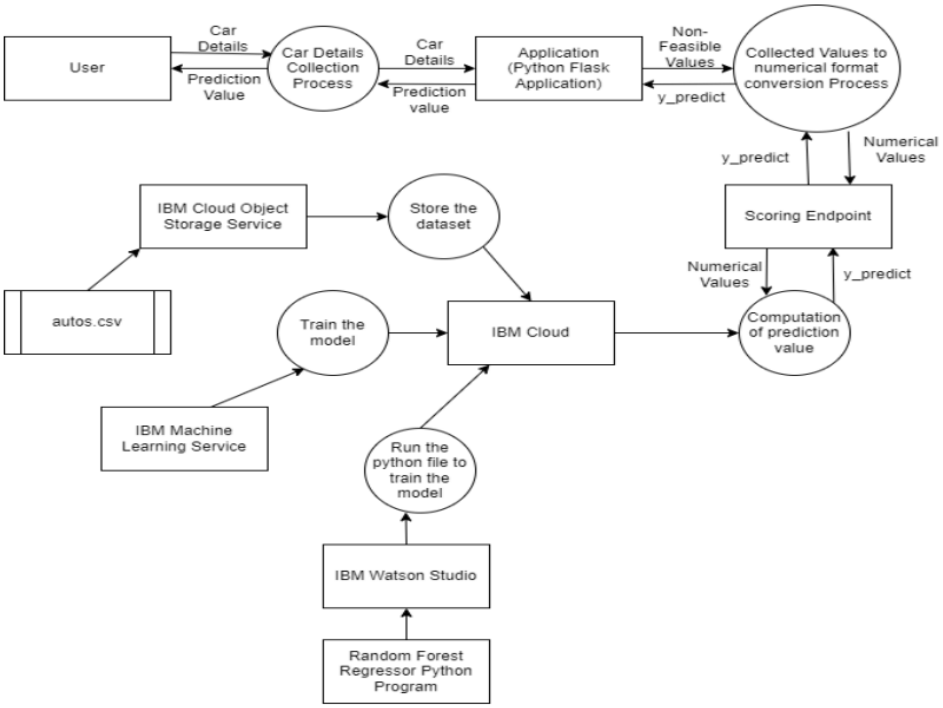
Level - 0



Level - 1

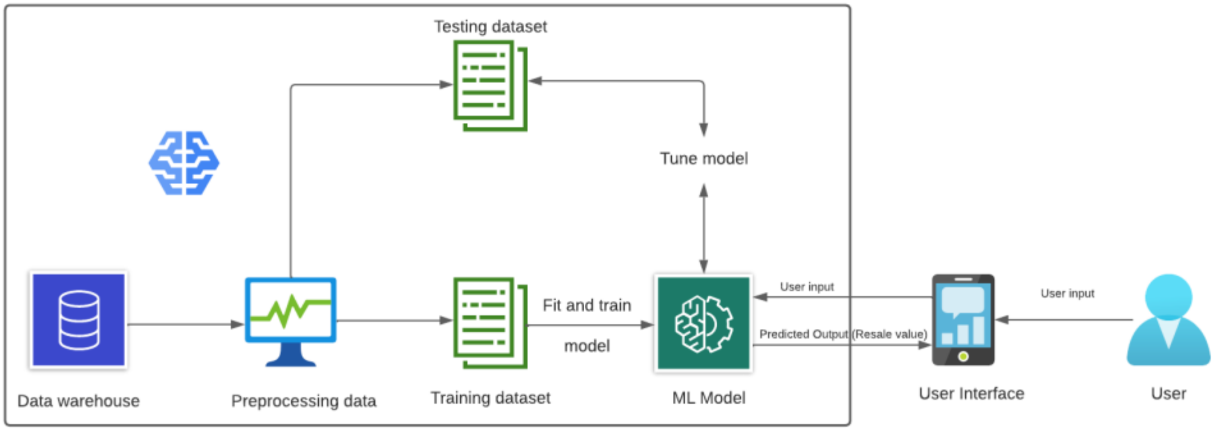

User
Car Details
Prediction Value
Car Details Collection Process
Car Details
Prediction value
Application (Python Flask Application)
Non-Feasible Values
y_predict
Collected Values to numerical format conversion Process
y_predict
Numerical Values
Scoring Endpoint
Numerical Values
y_predict
Random Forest Regressor Model
IBM Cloud
Computation of prediction value
autos.csv

Level - 2


User
Car Details
Prediction Value
Car Details Collection Process
Car Details
Prediction value
Application (Python Flask Application)
Non-Feasible Values
y_predict
Collected Values to numerical format conversion Process
y_predict
Numerical Values
Scoring Endpoint
Numerical Values
y_predict
IBM Cloud Object Storage Service
Store the dataset
autos.csv
Train the model
IBM Cloud
Computation of prediction value
IBM Machine Learning Service
Run the python file to train the model
IBM Watson Studio
Random Forest Regressor Python Program

5.2 Solution and Technical Architecture

Solution Architecture



Technical Interface

Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	The user interacts with application using Web UI	HTML, CSS, JavaScript etc.
2.	Application Logic-1	Logic for a process in the application	Python
3.	Database	The dataset containing car details is used for training the model to predict the rate	NoSQL.
4.	Cloud Database	The dataset is stored in the IBM cloud	IBM DB2.

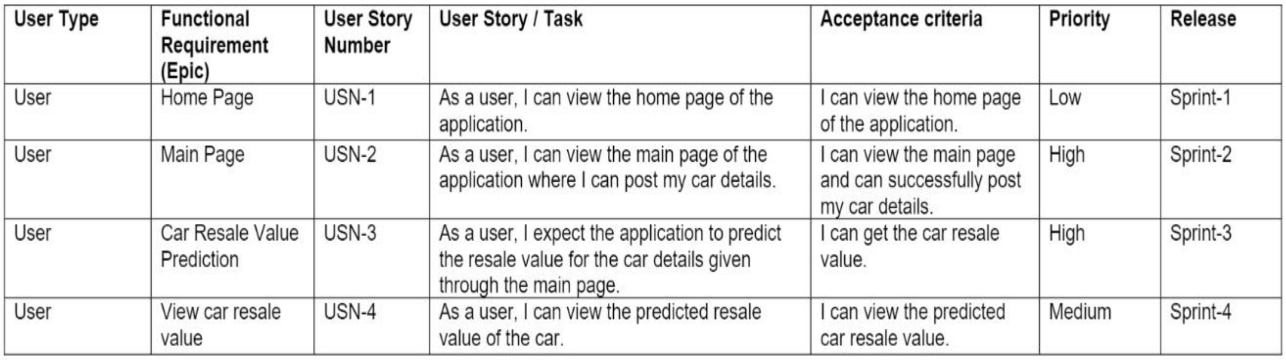
5.	File Storage	File storage requirements	IBM Block Storage.
6.	Machine Learning Model	It is responsible for predicting the resale value of the cars.	Regression Model
7.	Infrastructure (Server / Cloud)	Application will be deployed in cloud.	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S. No	**Characteristics**	**Description**	**Technology**
1.	Open-Source Frameworks	Open-source frameworks used	Flask, Python, IBM Cloud
2.	Security Implementations	Security / access controls implemented, use of firewalls etc.	Encryptions
3.	Scalable Architecture	Scalability of architecture consists of 3 tiers	Web Server - HTML, CSS, JavaScript Application Server - Python Flask Database Server - IBM Cloud
4.	Availability	User can access our application through cloud all the time	IBM Cloud Hosting.

5.	Performance	Multiple users can access the web application and can perform actions simultaneously	IBM Load Balance

5.3 User Stories



User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
User	Home Page	USN-1	As a user, I can view the home page of the application.	I can view the home page of the application.	Low	Sprint-1
User	Main Page	USN-2	As a user, I can view the main page of the application where I can post my car details.	I can view the main page and can successfully post my car details.	High	Sprint-2
User	Car Resale Value Prediction	USN-3	As a user, I expect the application to predict the resale value for the car details given through the main page.	I can get the car resale value.	High	Sprint-3
User	View car resale value	USN-4	As a user, I can view the predicted resale value of the car.	I can view the predicted car resale value.	Medium	Sprint-4

6.Project Planning and Scheduling

6.1 Sprint Planning and Estimation

TITLE	DESCRIPTION	DATE
Literature Survey & Information Gathering	Literature survey on the selected project and collecting other information	28 SEPTEMBER 2022
Prepare Empathy Map	Prepare Empathy Map Canvas to capture the user Pains & Gains, Prepare list of problem statements	19 SEPTEMBER2022

Ideation	List the by organizing the brainstorming session and prioritize the top 3 ideas based on the feasibility & importance.	19 SEPTEMBER2022
Proposed Solution	Prepare the proposed solution document, which includes the novelty, feasibility of idea, business model, social impact, scalability of solution, etc.	23 SEPTEMBER 2022
Problem Solution Fit	Prepare problem - solution fit document.	30 SEPTEMBER 2022
Solution Architecture	Prepare a solution architecture document.	28 SEPTEMBER 2022
Customer Journey	Prepare the customer journey maps to understand the user interactions & experiences with the application (entry to exit).	20 OCTOBER 2022
Functional Requirement	Prepare the functional requirement document.	8 OCTOBER 2022
Data Flow Diagrams	Draw the data flow diagrams and submit for review.	9 OCTOBER2022

Technology Architecture	Prepare the technology architecture diagram.	10 OCTOBER 2022
Prepare Milestone & Activity List	Prepare the milestones & activity list of the project.	22 OCTOBER 2022
Project Development - Delivery of Sprint-1, 2, 3 & 4	Develop & submit the developed code by testing it.	IN PROGRESS

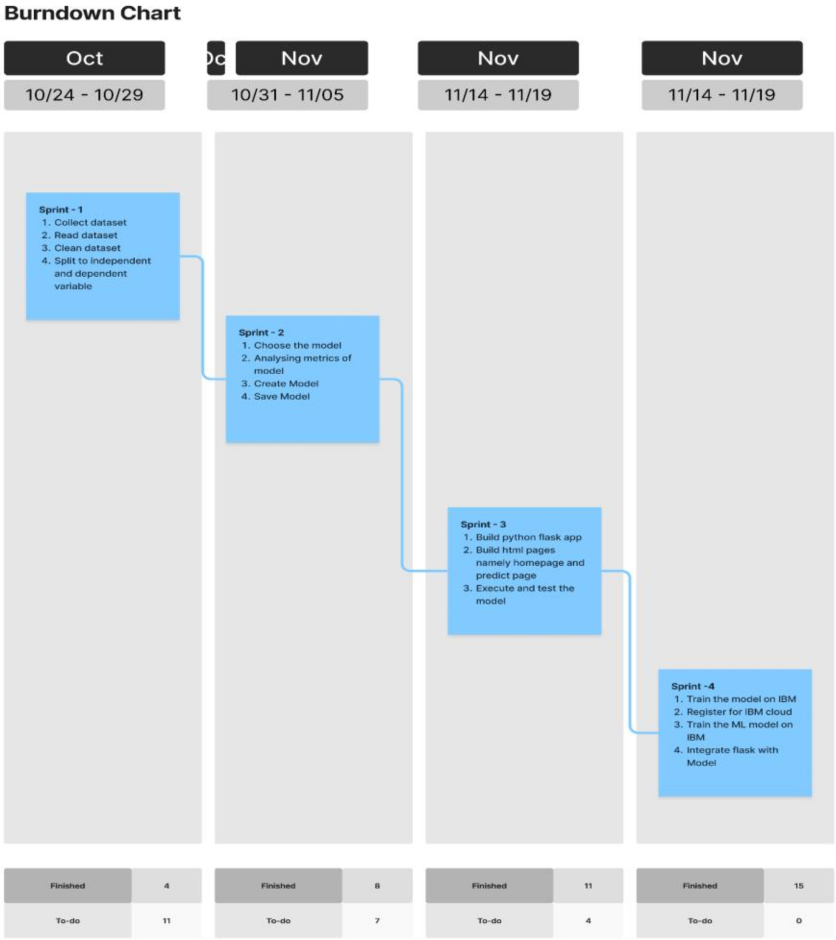
6.2 Sprint Delivery Schedule

Sprint	**Functional Requirement (Epic)**	**User Story Number**	**User Story / Task**	**Story points**	**Priorit y**	**Team Members**
Sprint-1	Collection and preprocessing the dataset		1. Collect dataset 2. Read dataset 3. Clean dataset 4. Split to independent and dependent variable	20	Low	Sidharth Abijeeth Vasra Arun Kalyan Kanishmithran Dinesh
Sprint-2	Model Building		1. Choose the model 2. Analyzing metrics of model 3. Create Model 4. Save Model	20	Mediu m	Sidharth Abijeeth Vasra Arun Kalyan Kanishmithran Dinesh
Sprint-3	Application Building	USN-1 USN-2 USN-3	1. Build python flask app 2. Build html pages namely homepage and predict page 3. Execute and test the model	20	High	Sidharth Abijeeth Vasra Arun Kalyan Kanishmithran Dinesh
Sprint-4	Training and Deployment	USN-4	1. Train the model on IBM 2. Register for IBM cloud 3. Train the ML model on IBM 4. Integrate flask with Model	20	Mediu m	Sidharth Abijeeth Vasra Arun Kalyan Kanishmithran Dinesh

Project Tracker, Velocity & Burndown Chart: (4 Marks)

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	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Burn-down Chart



Sprint - 1

1. Collect dataset
2. Read dataset
3. Clean dataset
4. Split to independent and dependent variable

Sprint - 2

1. Choose the model
2. Analysing metrics of model
3. Create Model
4. Save Model

Sprint - 3

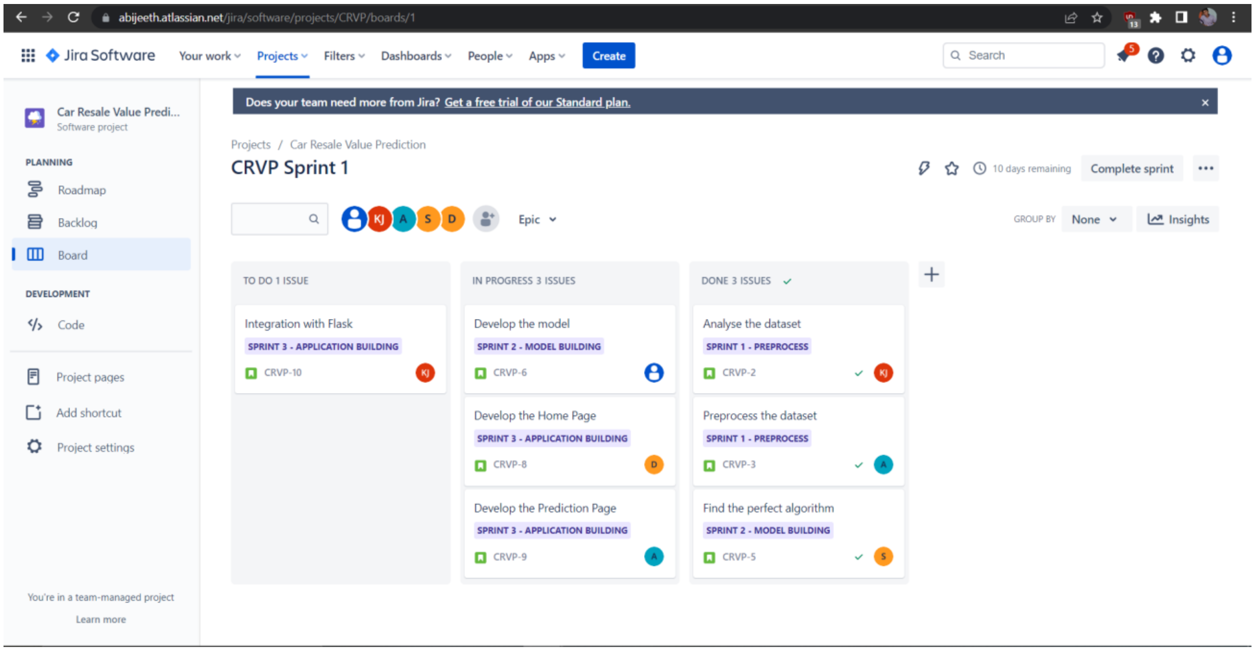
1. Build python flask app
2. Build html pages namely homepage and predict page
3. Execute and test the model

Sprint -4

1. Train the model on IBM
2. Register for IBM cloud
3. Train the ML model on IBM
4. Integrate flask with Model

Finished	4	Finished	8	Finished	11	Finished	15
To-do	11	To-do	7	To-do	4	To-do	0

6.3 Reports from JIRA



Does your team need more from Jira? Get a free trial of our Standard plan.

Projects / Car Resale Value Prediction

CRVP Sprint 1

TO DO 1 ISSUE

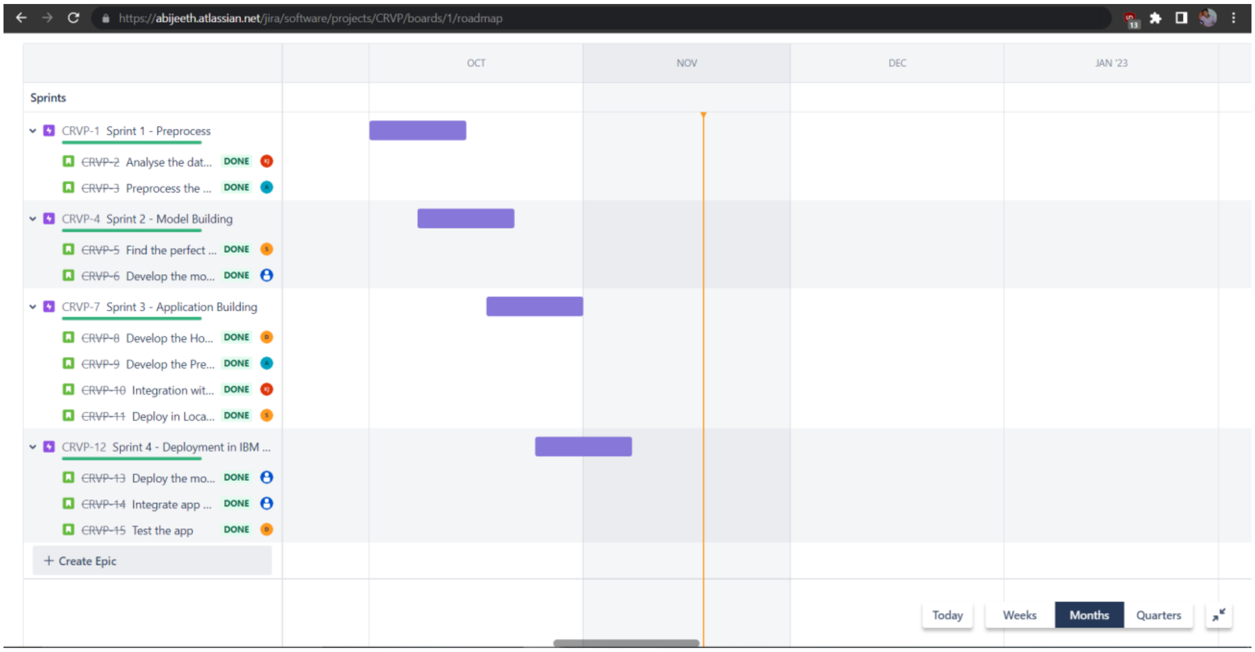
- Integration with Flask — SPRINT 3 - APPLICATION BUILDING — CRVP-10 — KJ

IN PROGRESS 3 ISSUES

- Develop the model — SPRINT 2 - MODEL BUILDING — CRVP-6
- Develop the Home Page — SPRINT 3 - APPLICATION BUILDING — CRVP-8 — D
- Develop the Prediction Page — SPRINT 3 - APPLICATION BUILDING — CRVP-9 — A

DONE 3 ISSUES ✓

- Analyse the dataset — SPRINT 1 - PREPROCESS — CRVP-2 ✓ KJ
- Preprocess the dataset — SPRINT 1 - PREPROCESS — CRVP-3 ✓ A
- Find the perfect algorithm — SPRINT 2 - MODEL BUILDING — CRVP-5 ✓ S

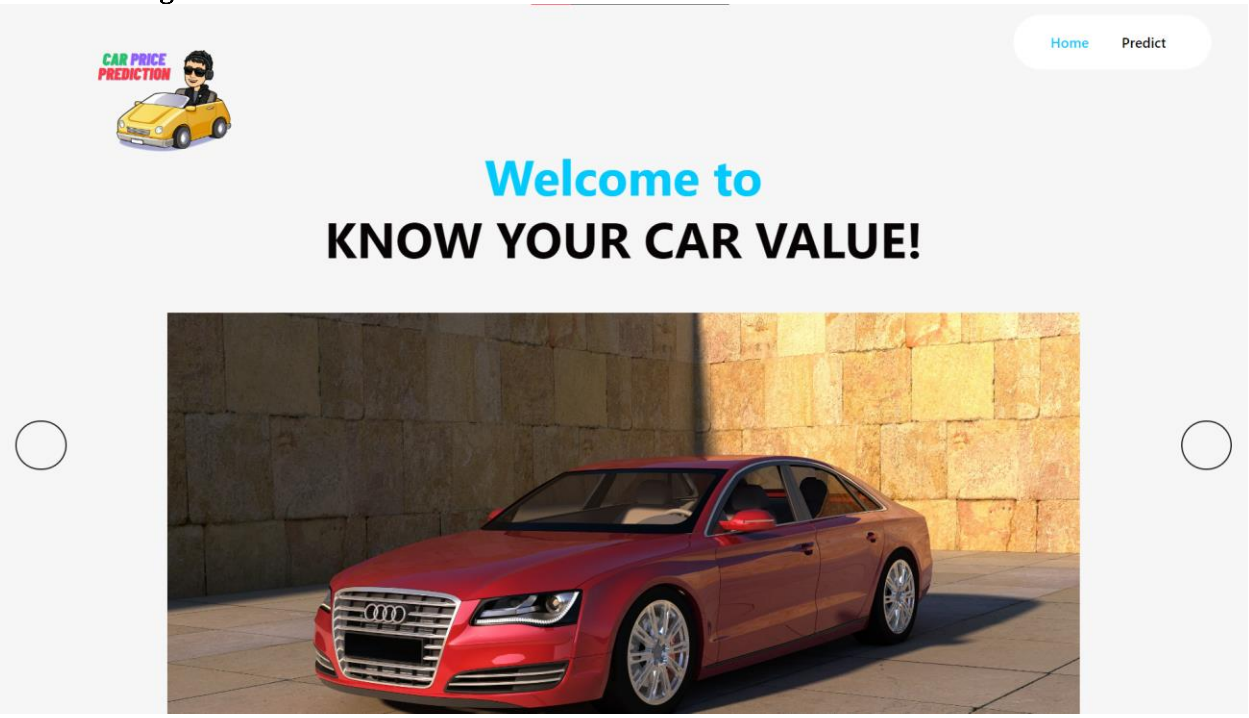

https://abijeeth.atlassian.net/jira/software/projects/CRVP/boards/1/roadmap
OCT
NOV
DEC
JAN '23
Today
Weeks
Months
Quarters
+ Create Epic

Sprints

- CRVP-1 Sprint 1 - Preprocess
  - ~~CRVP-2~~ Analyse the dat... DONE
  - ~~CRVP-3~~ Preprocess the ... DONE
- CRVP-4 Sprint 2 - Model Building
  - ~~CRVP-5~~ Find the perfect ... DONE
  - ~~CRVP-6~~ Develop the mo... DONE
- CRVP-7 Sprint 3 - Application Building
  - ~~CRVP-8~~ Develop the Ho... DONE
  - ~~CRVP-9~~ Develop the Pre... DONE
  - ~~CRVP-10~~ Integration wit... DONE
  - ~~CRVP-11~~ Deploy in Loca... DONE
- CRVP-12 Sprint 4 - Deployment in IBM ...
  - ~~CRVP-13~~ Deploy the mo... DONE
  - ~~CRVP-14~~ Integrate app ... DONE
  - ~~CRVP-15~~ Test the app DONE

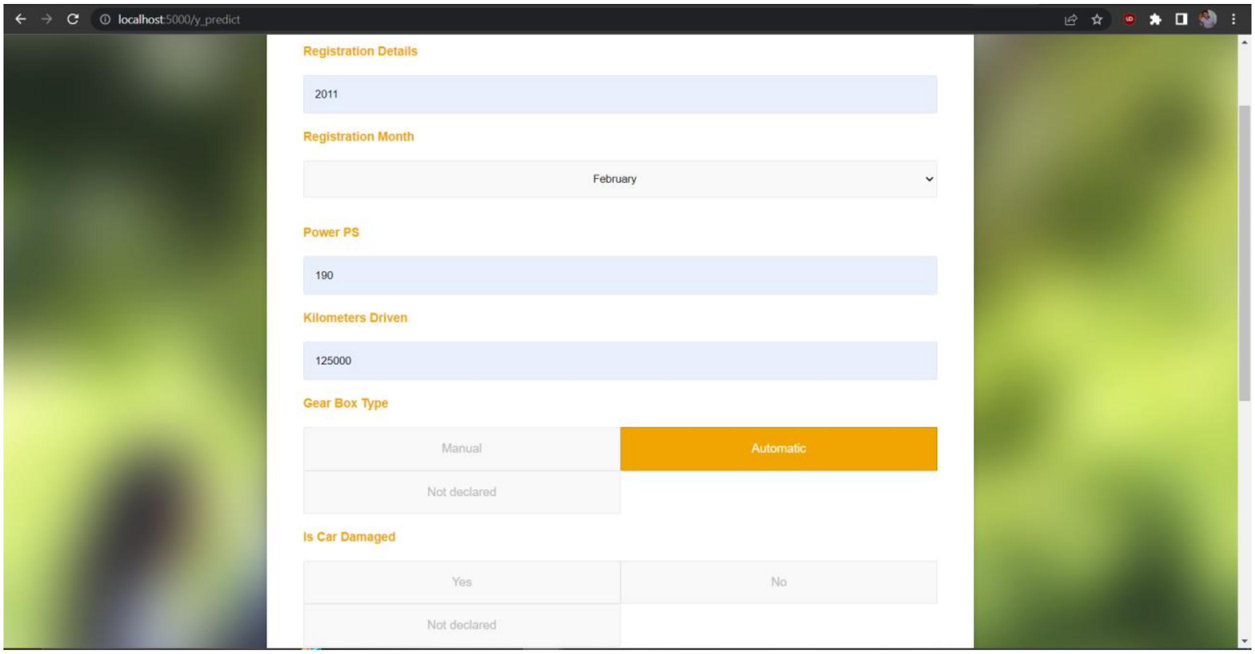
7.Coding and Solutioning

7.1 Home Page:



Welcome to
KNOW YOUR CAR VALUE!

7.2 Prediction Page



Registration Details

2011

Registration Month

February

Power PS

190

Kilometers Driven

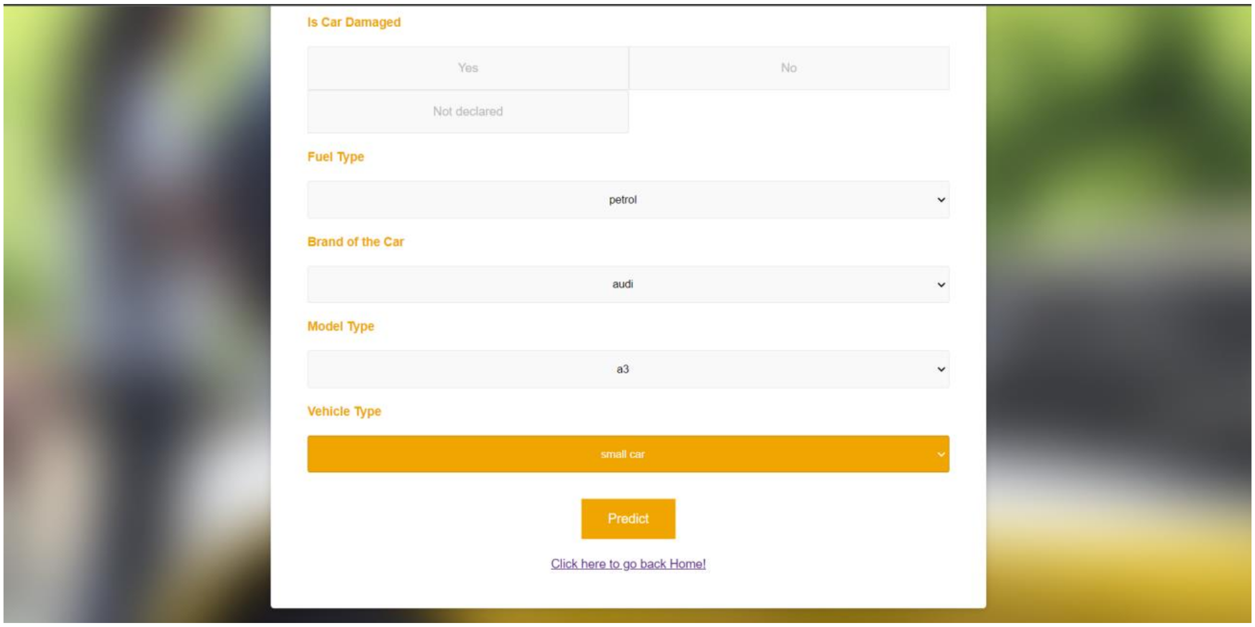
125000

Gear Box Type

Manual | Automatic | Not declared

Is Car Damaged

Yes | No | Not declared



Is Car Damaged

Fuel Type

petrol

Brand of the Car

audi

Model Type

a3

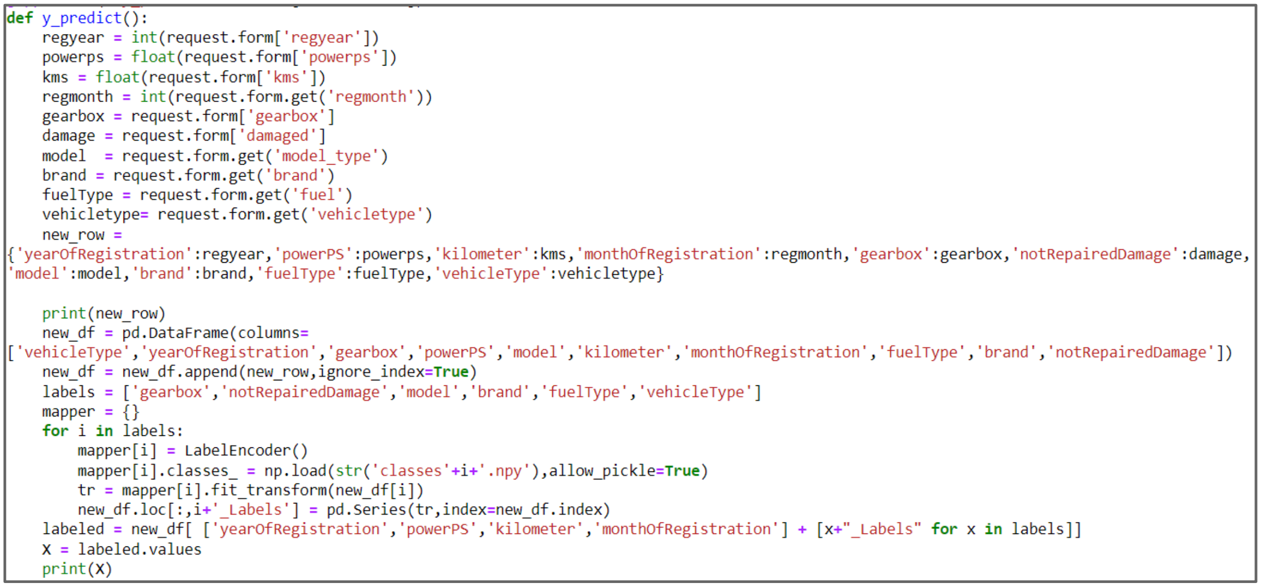
Vehicle Type

small car

Predict the Price!

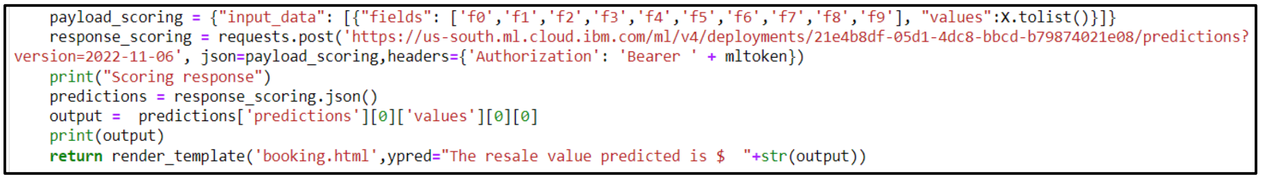
The resale value predicted is $ 19699.965761073345

- The user upon visiting the site will be displayed with a summary of what the project is all about, the user can get to know about the project and then go to the actual page.
- The page contains various fields of various types such as dropdown for categories, text fields for data, all the fields are made necessary.
- Once the user fills all the data and the predict button is selected, the data is transferred and the result is fetched, the result is converted in to Indian rupees for easy understanding.



```
def y_predict():
    regyear = int(request.form['regyear'])
    powerps = float(request.form['powerps'])
    kms = float(request.form['kms'])
    regmonth = int(request.form.get('regmonth'))
    gearbox = request.form['gearbox']
    damage = request.form['damaged']
    model  = request.form.get('model_type')
    brand = request.form.get('brand')
    fuelType = request.form.get('fuel')
    vehicletype= request.form.get('vehicletype')
    new_row = 
{'yearOfRegistration':regyear,'powerPS':powerps,'kilometer':kms,'monthOfRegistration':regmonth,'gearbox':gearbox,'notRepairedDamage':damage,
'model':model,'brand':brand,'fuelType':fuelType,'vehicleType':vehicletype}

    print(new_row)
    new_df = pd.DataFrame(columns=
['vehicleType','yearOfRegistration','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(str('classes'+i+'.npy'),allow_pickle=True)
        tr = mapper[i].fit_transform(new_df[i])
        new_df.loc[:,i+'_Labels'] = pd.Series(tr,index=new_df.index)
    labeled = new_df[ ['yearOfRegistration','powerPS','kilometer','monthOfRegistration'] + [x+"_Labels" for x in labels]]
    X = labeled.values
    print(X)
```

- From the above code snippet the user input is fetched from the user via request.form() method.
- A new data frame is created to send the values to the model to predict the price .
- The data is sent to the model in the IBM cloud to predict the value for the car



```
    payload_scoring = {"input_data": [{"fields": ['f0','f1','f2','f3','f4','f5','f6','f7','f8','f9'], "values":X.tolist()}]}
    response_scoring = requests.post('https://us-south.ml.cloud.ibm.com/ml/v4/deployments/21e4b8df-05d1-4dc8-bbcd-b79874021e08/predictions?
version=2022-11-06', json=payload_scoring,headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    predictions = response_scoring.json()
    output =  predictions['predictions'][0]['values'][0][0]
    print(output)
    return render_template('booking.html',ypred="The resale value predicted is $  "+str(output))
```

- The above code snippet is used to send the user inputs and fetch the output from the ML model present in IBM watson Studio.

- Payload Scoring sets the user inputs in the respective fields from f0 - f9
- Response Scoring is used to send the inputs to the IBM watson studio and fetch the output from the model

- Thus a render template is used to print the predicted value in the Page

8.Testing

8.1 Test Cases

Missing Values:

The trained ML model requires 4 feature inputs for predicting the output. Failing which, the model throws invalid Input error. All the fields in the html form have been marked required using CSS and thus the user must input all fields.

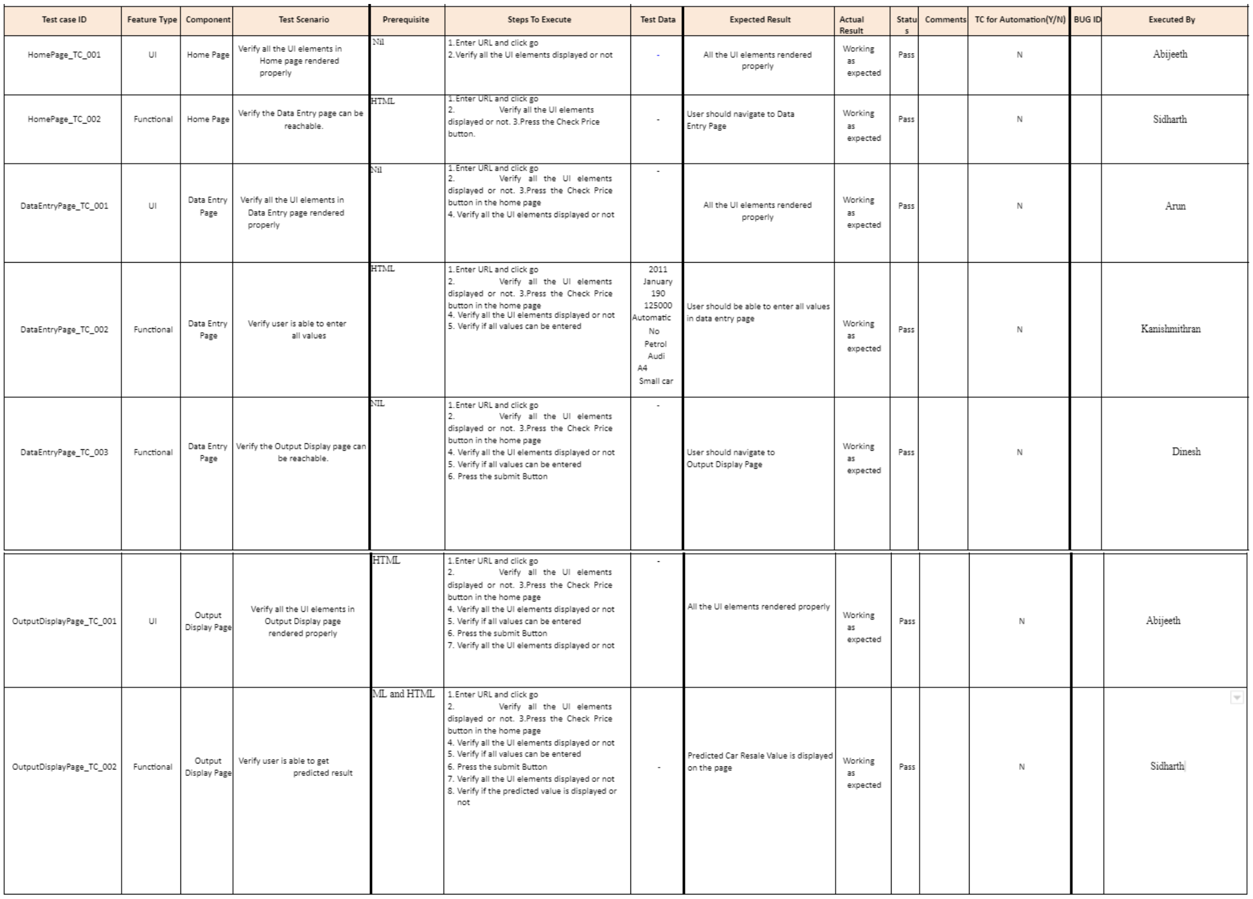
Output: User must input all the fields, failing which, form shows warning message "this field needs to be filled". Thus, there can be no errors in model prediction.

Invalid Input:

The trained ML model requires only numerical input for all 4 features. Thus, if the user uses symbols such as a comma while input, the model may throw an error. To overcome the same, preprocessing script is deployed in the backend which removes all unwanted characters like comma, whitespaces etc. so that model gets required input.

Output: Due to python preprocessing script, model will get the desired input and thus will give accurate prediction.

8.2 User Acceptance Testing

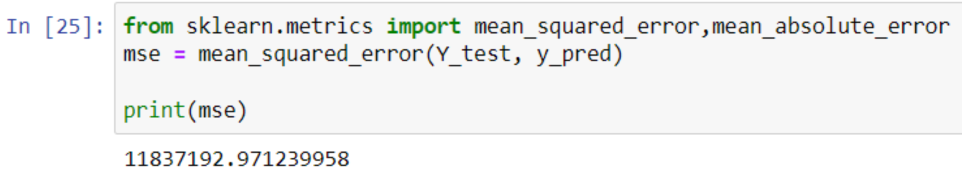


Test case ID	Feature Type	Component	Test Scenario	Prerequisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
HomePage_TC_001	UI	Home Page	Verify all the UI elements in Home page rendered properly	Nil	1.Enter URL and click go 2.Verify all the UI elements displayed or not	-	All the UI elements rendered properly	Working as expected	Pass		N		Abijeeth
HomePage_TC_002	Functional	Home Page	Verify the Data Entry page can be reachable.	HTML	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button.	-	User should navigate to Data Entry Page	Working as expected	Pass		N		Sidharth
DataEntryPage_TC_001	UI	Data Entry Page	Verify all the UI elements in Data Entry page rendered properly	Nil	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button in the home page 4. Verify all the UI elements displayed or not	-	All the UI elements rendered properly	Working as expected	Pass		N		Arun
DataEntryPage_TC_002	Functional	Data Entry Page	Verify user is able to enter all values	HTML	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button in the home page 4. Verify all the UI elements displayed or not 5. Verify if all values can be entered	2011 January 190 125000 Automatic No Petrol Audi A4 Small car	User should be able to enter all values in data entry page	Working as expected	Pass		N		Kanishmithran
DataEntryPage_TC_003	Functional	Data Entry Page	Verify the Output Display page can be reachable.	NIL	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button in the home page 4. Verify all the UI elements displayed or not 5. Verify if all values can be entered 6. Press the submit Button	-	User should navigate to Output Display Page	Working as expected	Pass		N		Dinesh
OutputDisplayPage_TC_001	UI	Output Display Page	Verify all the UI elements in Output Display page rendered properly	HTML	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button in the home page 4. Verify all the UI elements displayed or not 5. Verify if all values can be entered 6. Press the submit Button 7. Verify all the UI elements displayed or not	-	All the UI elements rendered properly	Working as expected	Pass		N		Abijeeth
OutputDisplayPage_TC_002	Functional	Output Display Page	Verify user is able to get predicted result	ML and HTML	1.Enter URL and click go 2. Verify all the UI elements displayed or not. 3.Press the Check Price button in the home page 4. Verify all the UI elements displayed or not 5. Verify if all values can be entered 6. Press the submit Button 7. Verify all the UI elements displayed or not 8. Verify if the predicted value is displayed or not	-	Predicted Car Resale Value is displayed on the page	Working as expected	Pass		N		Sidharth

9.Results

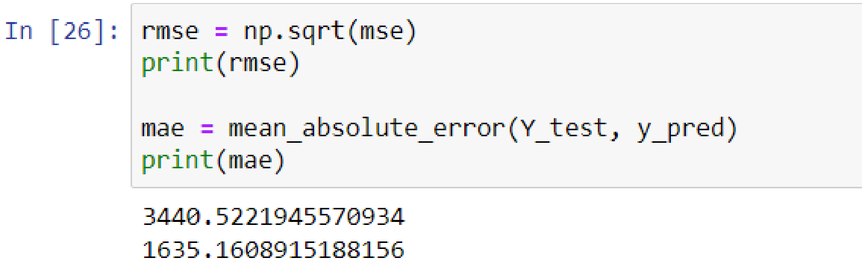
9.1 Performance Metrics

Mean Squared Error:



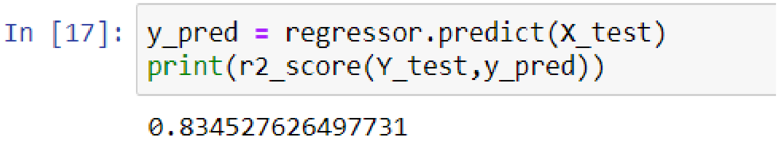
```
In [25]: from sklearn.metrics import mean_squared_error,mean_absolute_error
         mse = mean_squared_error(Y_test, y_pred)

         print(mse)

         11837192.971239958
```

Mean Absolute Error and Root Mean Squared Error:



```
In [26]: rmse = np.sqrt(mse)
         print(rmse)

         mae = mean_absolute_error(Y_test, y_pred)
         print(mae)

         3440.5221945570934
         1635.1608915188156
```

R2_score:



```
In [17]: y_pred = regressor.predict(X_test)
         print(r2_score(Y_test,y_pred))

         0.834527626497731
```

10.Advantages and Disadvantages

Advantages

- The website makes it simple for users to sell their used cars. Because the model's accuracy is close to 80%, they can obtain the ideal amount for their car.
- With the help of this site, the user can find the resale value of a car to buy or to sell one, without intermediaries, so they need not to spend money.

Disadvantages

- As a result of the smaller number of observations, the dataset was rather limited for drawing significant conclusions. More information gathered may result in more reliable predictions.
- There may be more characteristics that are reliable predictors. Here are some examples of variables that could enhance the model: doors, gas mileage (mpg), color, time spent undergoing mechanical and cosmetic repairs, used-to-new ratio, and appraisal-to-trade ratio.

11.Conclusion

Hence, various factors that affect the resale value of the vehicles were explored and the near accurate resale value of the vehicle is found out. Since, there has been an increase in the trends of buying vehicles, the proposed system will help find the accurate price of vehicles.

The proposed system reduces the time taken to predict the resale value of the vehicle and improves the efficiency of the model. It is made to be easily understood to the customers and highly reliable, hence no feature scaling is required.

12. Future Scope

In the future, we can introduce a login option, allowing us to save user searches and their personal information. Additionally, we can add a chat option that enables users to engage with sellers. So, Whenever a user is interested, they can communicate with the seller and can negotiate the price and take further actions.

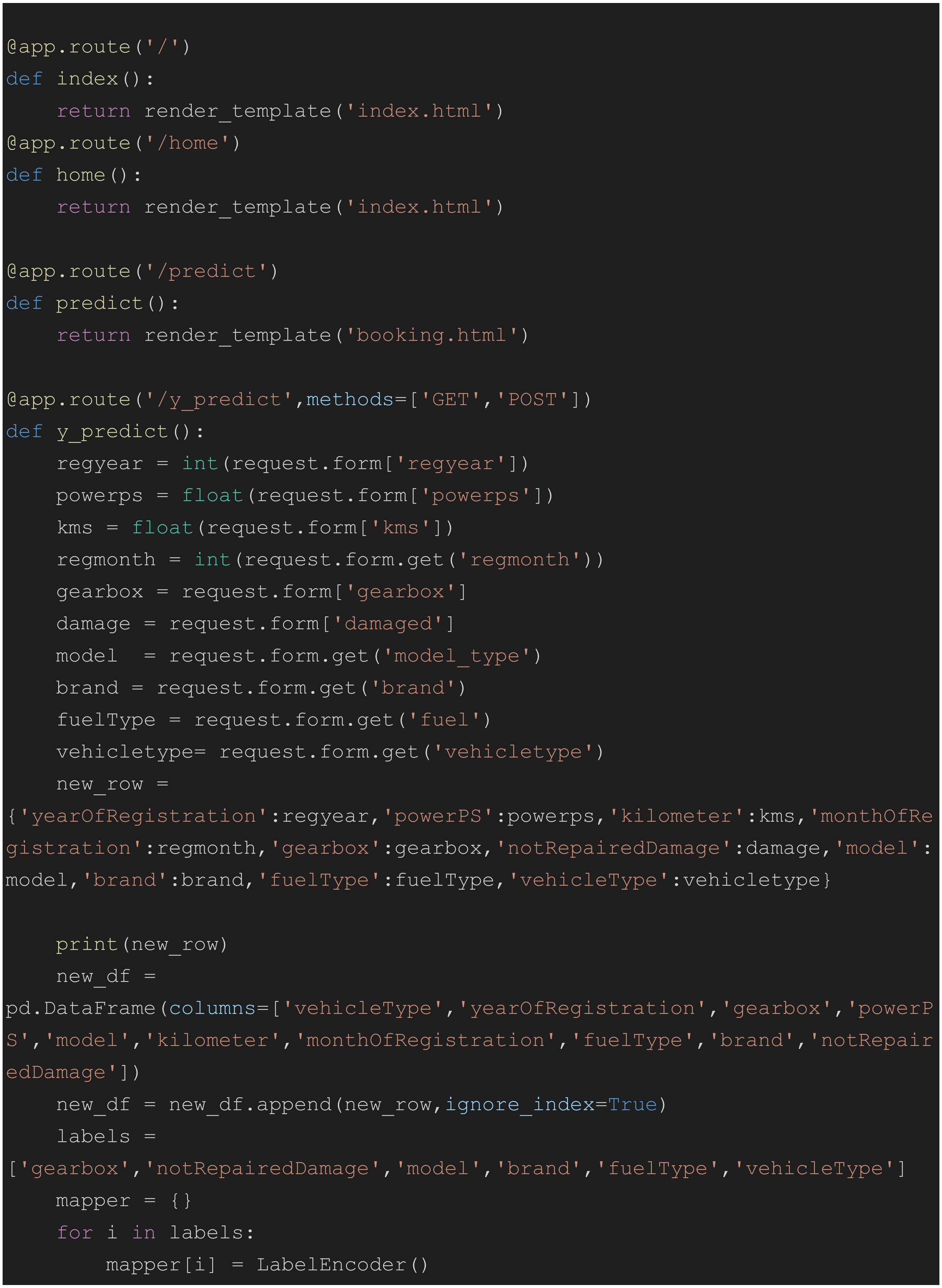
A VR model for the car can be added so that the user can get a 360-degree view of the car. In order to increase the performance, real time data shall be provided for price prediction and with historical data of car prices the accuracy of the system can be improved as well.

An android application could be developed in order to predict the resale value of the car as user interface for interacting with the customers who are not remotely available and for higher performance adaptive learning methodologies are taken into consideration and clusters of real time data are to be trained to predict the resale value.

13. Appendix

App.py:

```
import pandas as pd
import numpy as np
from flask import Flask,render_template,Response,request
import pickle
from sklearn.preprocessing import LabelEncoder
import pickle

import requests

# NOTE: you must manually set API_KEY below using information retrieved
from your IBM Cloud account.
API_KEY = "qwgXJFbBRzeK7VrTO9mnmGPjeNJOchWfASyjL-B6LO-U"
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__)
filename = 'resale_model.sav'
model_rand = pickle.load(open(filename,'rb'))
```



```
@app.route('/')
def index():
    return render_template('index.html')
@app.route('/home')
def home():
    return render_template('index.html')

@app.route('/predict')
def predict():
    return render_template('booking.html')

@app.route('/y_predict',methods=['GET','POST'])
def y_predict():
    regyear = int(request.form['regyear'])
    powerps = float(request.form['powerps'])
    kms = float(request.form['kms'])
    regmonth = int(request.form.get('regmonth'))
    gearbox = request.form['gearbox']
    damage = request.form['damaged']
    model  = request.form.get('model_type')
    brand = request.form.get('brand')
    fuelType = request.form.get('fuel')
    vehicletype= request.form.get('vehicletype')
    new_row =
{'yearOfRegistration':regyear,'powerPS':powerps,'kilometer':kms,'monthOfRe
gistration':regmonth,'gearbox':gearbox,'notRepairedDamage':damage,'model':
model,'brand':brand,'fuelType':fuelType,'vehicleType':vehicletype}

    print(new_row)
    new_df =
pd.DataFrame(columns=['vehicleType','yearOfRegistration','gearbox','powerP
S','model','kilometer','monthOfRegistration','fuelType','brand','notRepair
edDamage'])
    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(str('classes'+i+'.npy'),allow_pickle=True)
        tr = mapper[i].fit_transform(new_df[i])
        new_df.loc[:,i+'_Labels'] = pd.Series(tr,index=new_df.index)
    labeled = new_df[
['yearOfRegistration','powerPS','kilometer','monthOfRegistration'] +
[x+"_Labels" for x in labels]]
    X = labeled.values
    print(X)
    #y_prediction = model_rand.predict(X)
    #print(y_prediction)
    # NOTE: manually define and pass the array(s) of values to be scored
in the next line
    payload_scoring = {"input_data": [{"fields":
['f0','f1','f2','f3','f4','f5','f6','f7','f8','f9'],
"values":X.tolist()}]}
    response_scoring = requests.post('https://us-
south.ml.cloud.ibm.com/ml/v4/deployments/21e4b8df-05d1-4dc8-bbcd-
b79874021e08/predictions?version=2022-11-06',
json=payload_scoring,headers={'Authorization': 'Bearer ' + mltoken})
    print("Scoring response")
    predictions = response_scoring.json()
    output =  predictions['predictions'][0]['values'][0][0]
    print(output)
    return render_template('booking.html',ypred="The resale value
predicted is $  "+str(output))

if __name__ == '__main__':
    app.run(host='Localhost',debug=True,threaded=False)
```

Links

GitHub Link	https://github.com/IBM-EPBL/IBM-Project-32310-1660209166
Project Demo Link	www.shorturl.at/ftyNV