Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies

TEAM ID: PNT2022TMID27220

1. INTRODUCTION:

1.1Project overview:

Today's world is seeing a substantial increase in automobiles. Because there are more cars on the road and more people are driving them at high speeds, accidents happen more frequently. When an accident occurs, the parties involved file a claim with their auto insurance to obtain the money needed to repair the vehicle since, according to false claims, the company acts inappropriately and withholds payments. Claims leakage, or the gap between the amounts the firm has secured and the amounts it should have secured in line with the claims, is the cause of this. The claim process will take longer than usual in accordance with business policy, even though the car's damage is clearly visible. Despite the company's best efforts, the claim's processing is taking longer than expected. Differentiate the suggested strategy to maybe expedite the evaluation of car damage. When given a picture of a damaged vehicle, a system may be able to complete automobile damage detection in a matter of seconds rather than taking hours to do so if it were visually evaluated. The technology is able to ascertain the analysis of utilising computer vision and machine learning to determine the extent of the damage, its location, and its size.

1.2purpose:

In the modern world, the number of automobiles has significantly increased. Accidents occur more frequently because there are more cars on the road and more individuals are driving them fast. When an accident occurs, the parties involved file a claim with their auto insurance to get the money needed to fix the car because, based on false claims, the insurance provider behaves improperly and withholds payments.

2. LITERATURE SURVEY:

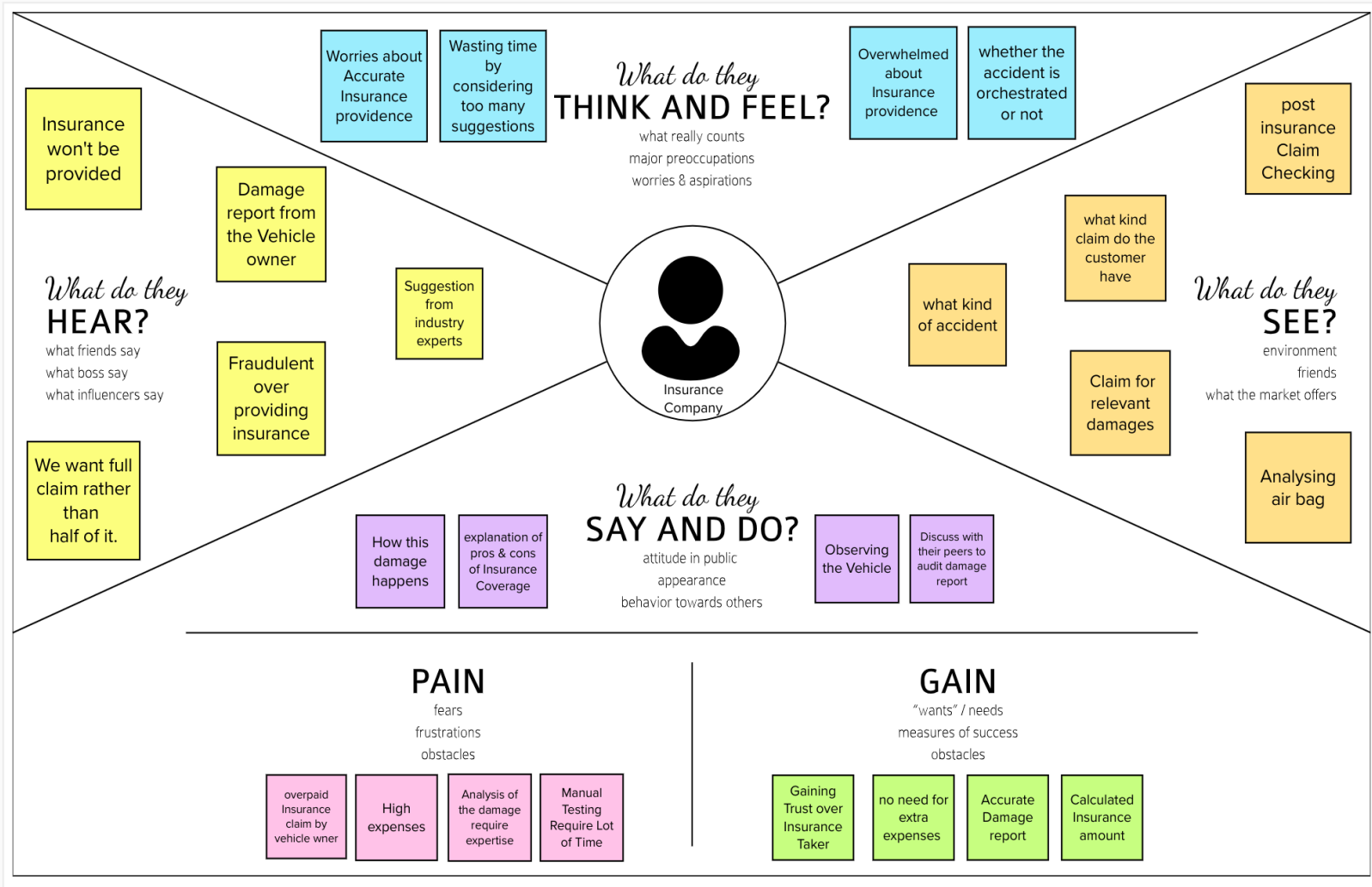
S NO.	TITLE	AUTHORS	ABSTRACT	DRAWBACKS
1.	Research on Intelligent Vehicle Damage Assessment System Based on Computer Vision	Zhu Qianqian ,Guo Weiming ,ShenYing and ZhaoZihao	In this paper, based on the demand of automobile insurance claims for intelligent transportation, combined with abundant basic data and advanced machine vision algorithms, an intelligent damage determination system of 'Artificial Intelligence Vehicle Insurance' is constructed. This paper first introduces the functions of the intelligent damage assessment system.Secondly,it discusses the realization path of each functional module in detail, and finally puts forward the vision for the future.	The drawback is to explore the innovation of insurance technology of 'AI + Vehicle Insurance.

2.	Damage Assessment of a vehicle and Insurance Reclaim.	Vaibhav Agarwal ,Utsav Khandelwal, Shivam Kumar, Raja Kumar, Shilpa M	By reducing loss adjustment costs, improvements in the First Notice of Loss and the speed with which claims are examined and evaluated might save a lot of money in the automobile insurance claims process. Car damage is automatically identified and classified using advanced picture analysis and pattern recognition technology. A technique that compares before-and after-accident car images to automatically detect the damaged location.	The major Drawback of the Proposed model is that it only Identifies the Physical visible Damage and not Of the internal or the interior damage.
3.	Assessing Car Damage with Convolutional Neural Networks	Harit Bandi,Suyash Joshi,Siddhant Bhagat,Amol Deshpande	Manual estimation of damages in fields like construction, vehicular accidents has been the mainstay of the insurance business. However, such methods are replete with biases and inaccurate estimations. This paper deals with estimating car damage, primarily with auto insurers as our key potential customers. For this purpose, three distinct Transfer Learning approaches are used which detect the presence of damage, location, and severity of the damage.	The drawback here is Driver behavior monitoring.Machine learning enhanced solutions help in monitoring driver's behavior.

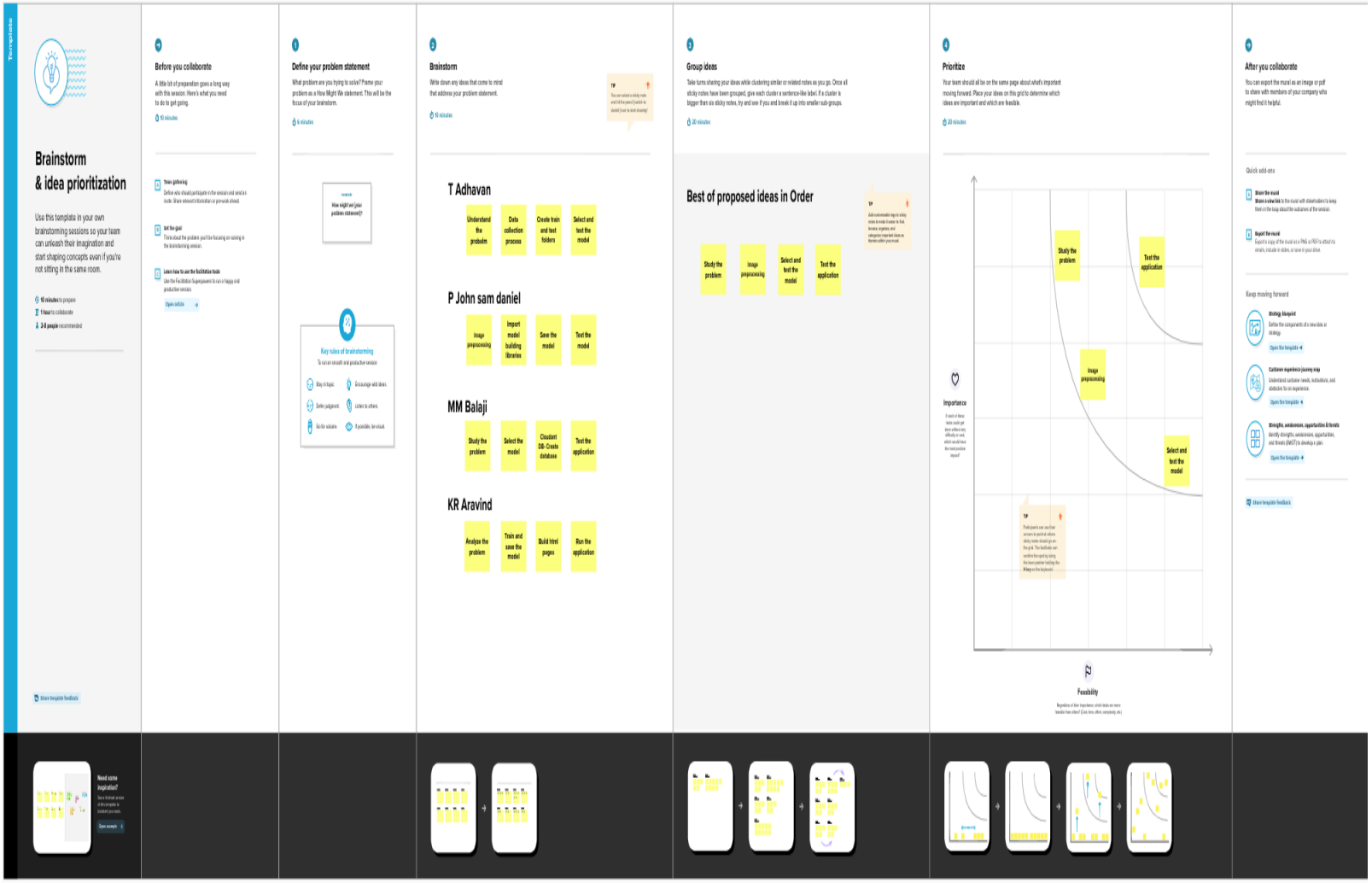
4.	Car Damage Assessment for Insurance Companies	Mandara G and Prashant Ankalkoti	The data contains three classes namely train, test and validation. Trained image is compared with the test image. Car has to be trained for many times by using epochs which means how many times the algorithm can work between the whole training dataset. In this graph they can take only two times of running the algorithm.Finally the comparison is completed lastly print the graph containing accuracy, validation accuracy, loss and validation loss.	Need for human involvement. Although the process could be absolutely automated, it still needs human involvement to detect and avoid fraudulent insurance cases.
5.	Digital Transformation in Car Insurance Industry: Streamline Recognition of Car Damage Assessment	Max Galaktionov	Digital transformation and Machine Learning technologies enable automation which is actively been used in the car insurance industry. It enables quick vehicle damage detection, improves management, cuts employee expenses, and allows to improve the overall quality of service.	The challenge is Processing of big volumes of data. The insurers need to be able to quickly assess and analyze data from various sources and provide exact estimations.

3. IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map:



3.2 Ideation and Brainstorming:



3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Intelligent vehicle damage assessment and cost estimator for insurance companies.
2.	Idea / Solution description	To perceive and determine the precise degree of automotive damage, we are building an AI model.
3.	Novelty / Uniqueness	Calculator for filing an insurance claim that is automated.
4.	Social Impact / Customer Satisfaction	Estimating the degree of car damage and providing insurance in accordance.
5.	Business Model (Revenue Model)	The primary sources of income for insurers are underwriting and investment income. The majority of an insurance company's assets are financial investments, such as listed shares, government bonds, commercial real estate, and corporate bonds. They are able to save more money and reinvest it in their own businesses by estimating the extent of car damage using our AI model and giving insurance in accordance.
6.	Scalability of the Solution	Our artificial intelligence (AI) can function at the scale, speed, and complexity necessary to achieve the goal. As our model is tested and trained using additional real-time data, its accuracy will rise.

3.4 PROBLEM SOLUTION FIT

Project Title: Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies

Project Design Phase-I - Solution Fit Template

Team ID: PNT2022TMID27220

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) — CS

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

I. A commercial worker moving from one location to another.
II. People belonging over the age of 18
III. Person whose vehicle was involved in an accident or sustained damage
IV. An insured consumer who can make a claim

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.

I. Unreliable network connections could make some functions inaccessible.
II. Improper images or blurred images might affect the accurate performance of the application.

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

I. requesting an estimate of costs from a third party
II. Manual computations are used for cost estimation.
III. Using computationally expensive algorithms to find the harm.

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.

I. The primary issue will be the amount of time required to calculate the cost and percentage of damage.
II. Addressing issues like this is very important identify and unify exact damage ratios cost of this damage
III. The corporations did not perfectly compensate for the damage.

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.

I. Deviation or difference between the cost calculated by the company and the actual cost
II. Many advanced methods of estimating have emerged as a result of the AI field's quick development.
III. Customers must do it as a result of the new regulations.

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)

I. After an accident, the customer must upload pictures of the vehicle.
II. The software will evaluate the damages immediately and present the claim amount to the users.

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.

I. Advancing technology in the field of forecasting and estimation.
II. coworkers and society calling for a quick insurance claim
III. Customer seeking independence and avoiding false traps

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.

I. Determine the damage percentage with precision.
II. As it relates to the car, forecast the area of damage.
III. For functionality, employ quick processing algorithms.

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.

8.1 ONLINE

I. You can browse a webpage to estimate damage using an input image.
II. Rapid access to the damage assessment method based on artificial intelligence.

Identify strong TR & EM

4. REQUIREMENTS 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 Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Interface	Login System,Dashboard,Uploading Image,Review and Analyze the results.
FR-4	Collection of datasets	Information about the user and their vehicle. Information about Insurance plans.
FR-5	Results	The model must be structured with high accuracy. The results obtained from the model will be displayed for the user to understand easily.

4.2 Non-functional Requirements:

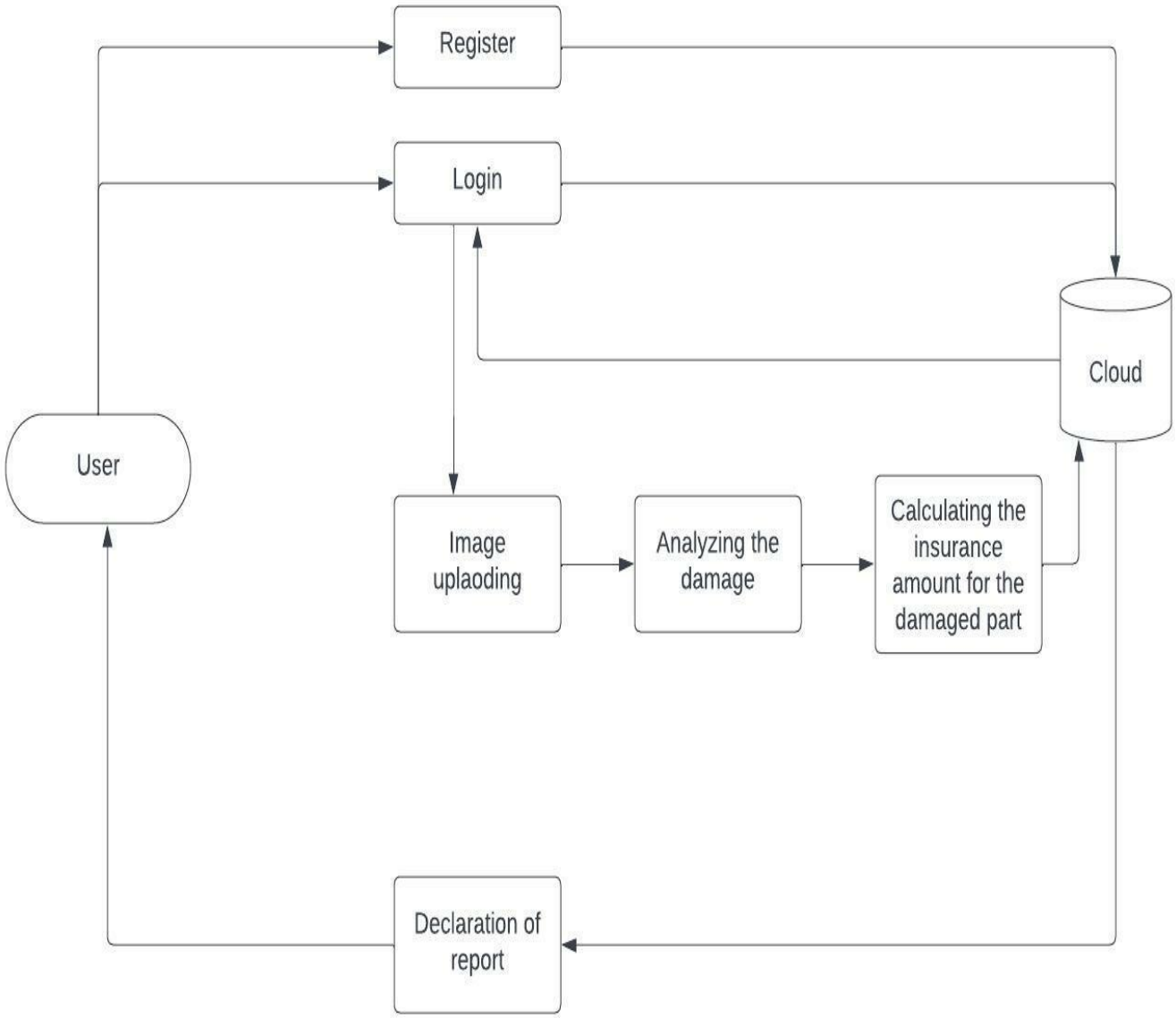
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Intelligent model for damage assessment in vehicle and cost estimate provided by insurance company.
NFR-2	**Security**	The authenticity of the user and the confidentiality of the user's details relating to his vehicle must be preserved.
NFR-3	**Reliability**	This project needs to achieve good accuracy in damage assessment as well as cost estimation so that users receive an accurate and unbiased amount of insurance.

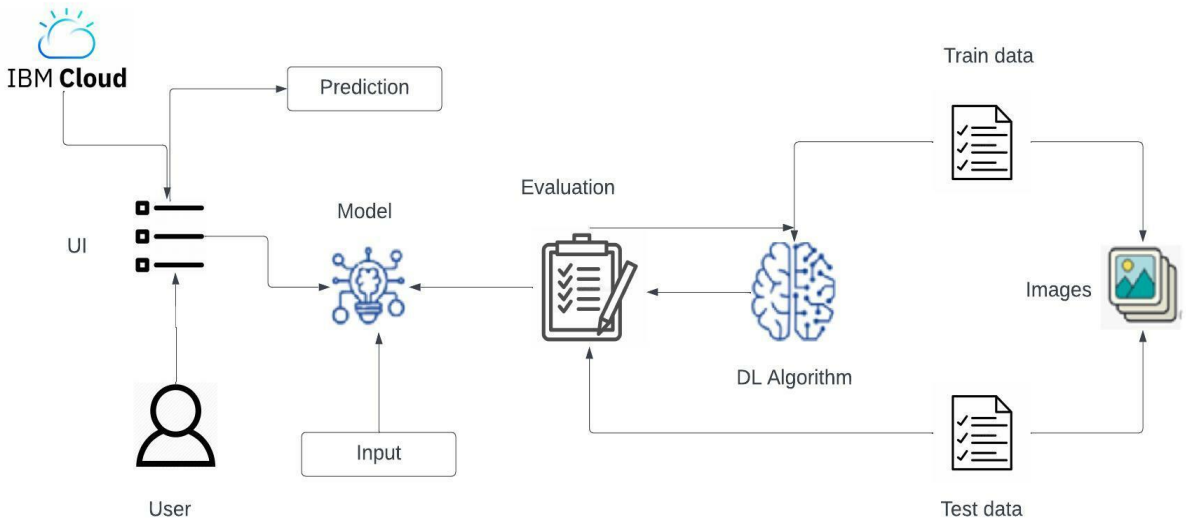
NFR-5	**Availability**	The webpage must be compatible with web browsers on mobile phones and computers.

5. PROJECT DESIGN:

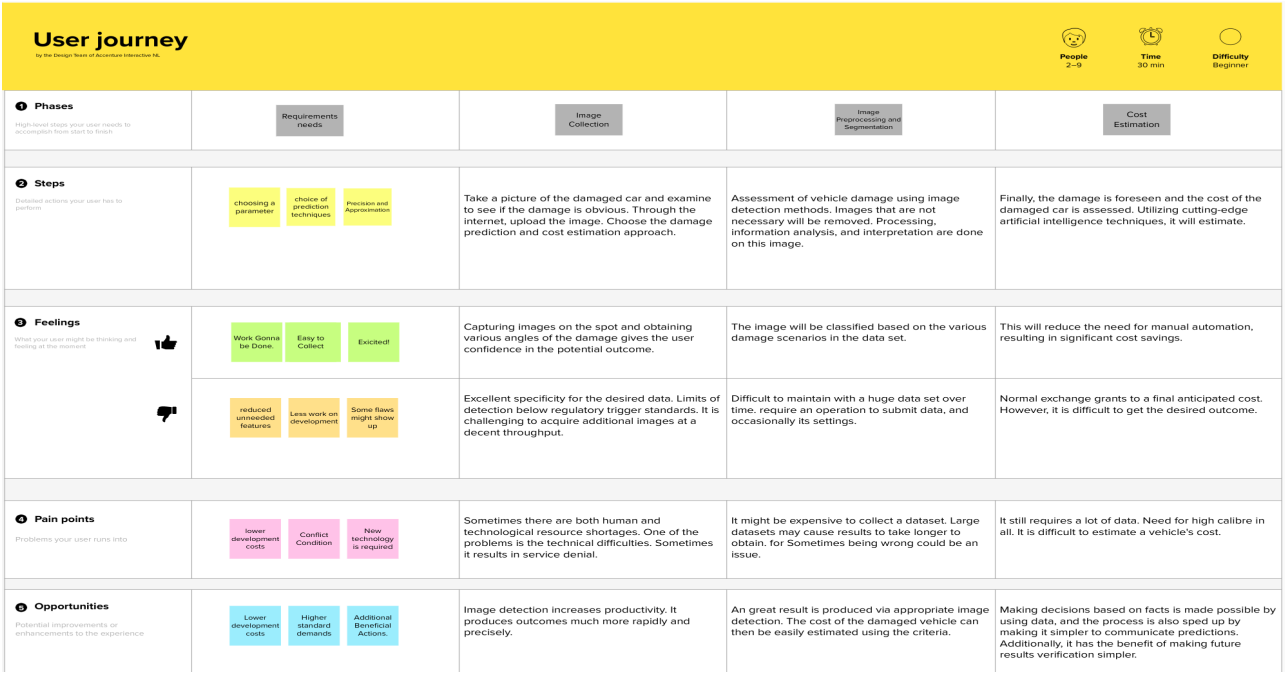
5.1 Data flow diagram:



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES



User journey

by the Design Team of Accenture Interactive NL

People 2–9 | Time 30 min | Difficulty Beginner

	Requirements needs	Image Collection	Image Preprocessing and Segmentation	Cost Estimation
1 Phases High-level steps your user needs to accomplish from start to finish				
2 Steps Detailed actions your user has to perform	choosing a parameter; choice of prediction techniques; Precision and Approximation	Take a picture of the damaged car and examine to see if the damage is obvious. Through the internet, upload the image. Choose the damage prediction and cost estimation approach.	Assessment of vehicle damage using image detection methods. Images that are not necessary will be removed. Processing, information analysis, and interpretation are done on this image.	Finally, the damage is foreseen and the cost of the damaged car is assessed. Utilizing cutting-edge artificial intelligence techniques, it will estimate.
3 Feelings What your user might be thinking and feeling at the moment	Work Gonna be Done.; Easy to Collect; Excited!	Capturing images on the spot and obtaining various angles of the damage gives the user confidence in the potential outcome.	The image will be classified based on the various damage scenarios in the data set.	This will reduce the need for manual automation, resulting in significant cost savings.
	reduced unneeded features; Less work on development; Some flaws might show up	Excellent specificity for the desired data. Limits of detection below regulatory trigger standards. It is challenging to acquire additional images at a decent throughput.	Difficult to maintain with a huge data set over time. require an operation to submit data, and occasionally its settings.	Normal exchange grants to a final anticipated cost. However, it is difficult to get the desired outcome.
4 Pain points Problems your user runs into	lower development costs; Conflict Condition; New technology is required	Sometimes there are both human and technological resource shortages. One of the problems is the technical difficulties. Sometimes it results in service denial.	It might be expensive to collect a dataset. Large datasets may cause results to take longer to obtain. for Sometimes being wrong could be an issue.	It still requires a lot of data. Need for high calibre in all. It is difficult to estimate a vehicle's cost.
5 Opportunities Potential improvements or enhancements to the experience	Lower development costs; Higher standard demands; Additional Beneficial Actions.	Image detection increases productivity. It produces outcomes much more rapidly and precisely.	An great result is produced via appropriate image detection. The cost of the damaged vehicle can then be easily estimated using the criteria.	Making decisions based on facts is made possible by using data, and the process is also sped up by making it simpler to communicate predictions. Additionally, it has the benefit of making future results verification simpler.

6. PROJECT PLANNING & SCHEDULING:

6.1 SPRINT PLANNING & ESTIMATION:

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	6	6 Days	24 Oct 2022	29 Oct 2022	5	29 Oct 2022
Sprint-2	6	6 Days	31 Oct 2022	05 Nov 2022	5	05 Nov 2022
Sprint-3	6	6 Days	07 Nov 2022	12 Nov 2022	5	12 Nov 2022
Sprint-4	6	6 Days	14 Nov 2022	19 Nov 2022	5	19 Nov 2022

6.2 SPRINT DELIVERY SCHEDULE:

Sprint	Duration	Sprint Start Date	Sprint End Date (Planned)	Sprint Release Date (Actual)
Sprint-1	6 Days	24 Oct 2022	29 Oct 2022	29 Oct 2022
Sprint-2	6 Days	31 Oct 2022	05 Nov 2022	05 Nov 2022
Sprint-3	6 Days	07 Nov 2022	12 Nov 2022	12 Nov 2022
Sprint-4	6 Days	14 Nov 2022	19 Nov 2022	19 Nov 2022

6.3 REPORTS FROM JIRA:

Velocity:

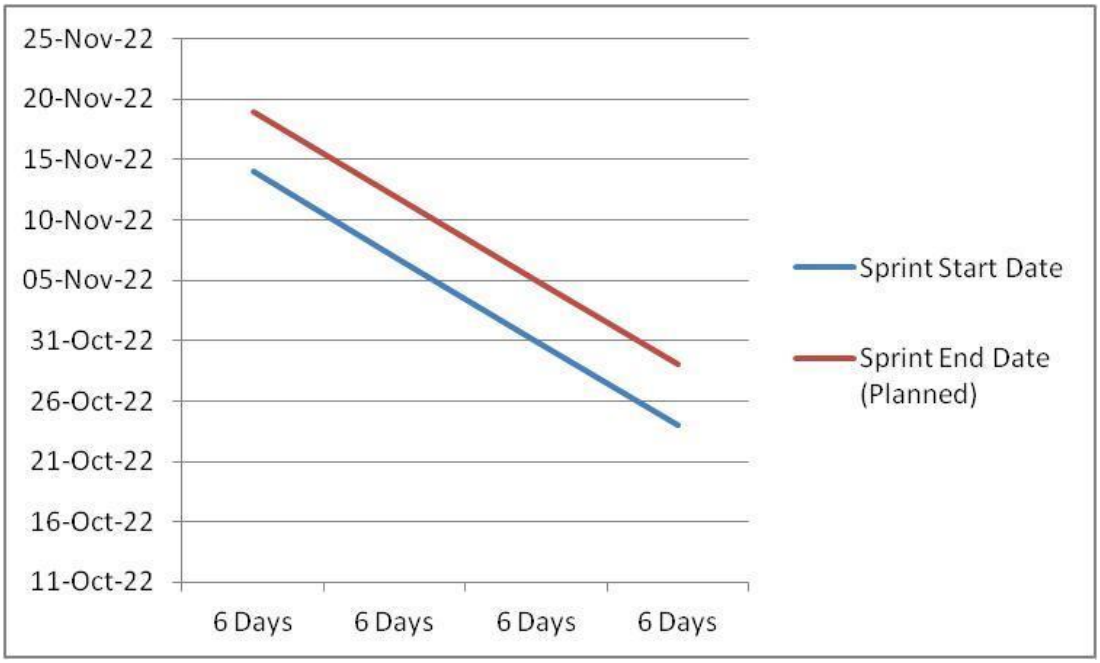
We have a 10-day sprint duration. The velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

AV = Sprint duration/Velocity

= 6/6

= 1

Burn down Chart:

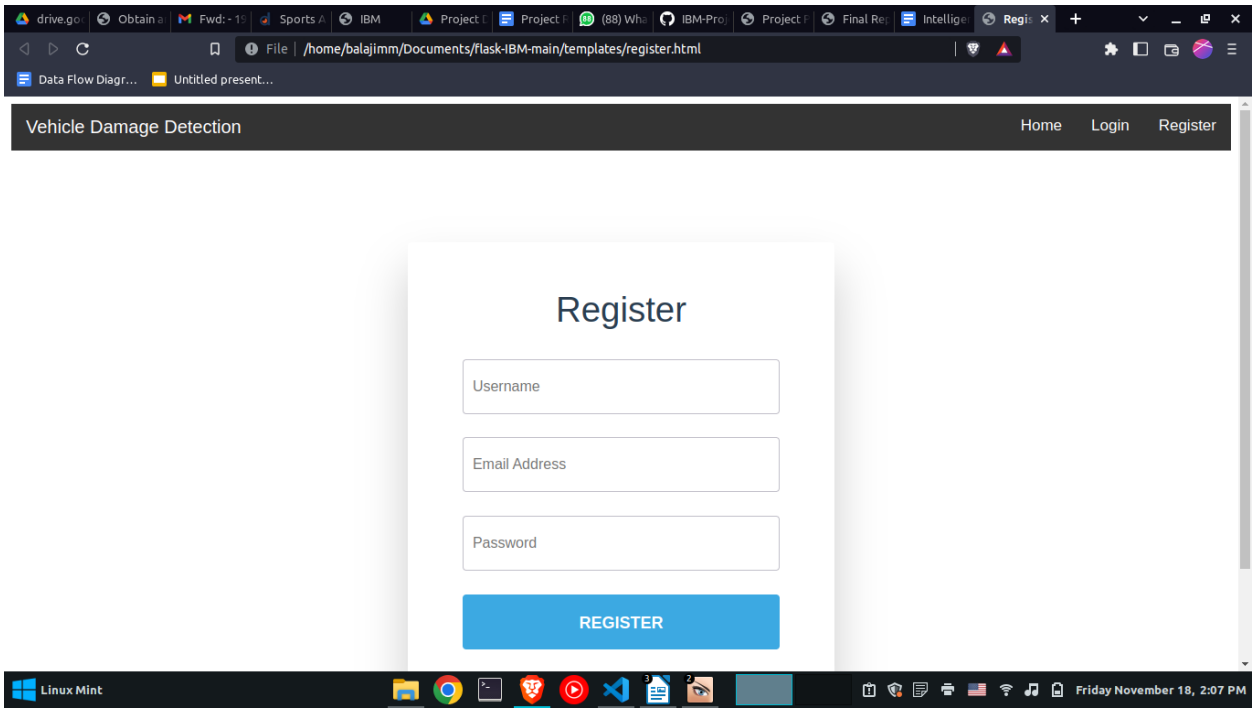


7. CODING & SOLUTIONING:

New user:

```
<!DOCTYPE html>
<html lang="en" >
<head>
 <meta charset="UTF-8">
 <title>Register</title>
 <!-- <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/normalize/5.0.0/normalize.min
.css">
<link rel="stylesheet" href="E:\IBM\static\styles\Register.css"> -->
</head>
<body>
 <div class="topnav">
   <a  id="logo" href=""> Vehicle Damage Detection</a>
   <a href="{{ url_for('register') }}">Register</a>
   <a href="{{ url_for('login') }}">Login</a>
   <a href="{{ url_for('index') }}">Home</a>
 </div>
<div id="login-form-wrap">
 <h2>Register</h2>
 <form id="login-form" method="POST" action="/afterreg">
   <p>
       <input type="text" id="text" name="name" placeholder="Username"
required><i class="validation"><span></span><span></span></i>
   </p>
   <p>
       <input type="email" id="email" name="email" placeholder="Email
Address" required><i class="validation"><span></span><span></span></i>
   </p>
   <p>
       <input type="password" id="password" name="psw"
placeholder="Password" required=""><i
class="validation"><span></span><span></span></i>
   </p>
   <p>
   <input type="submit" id="login" value="Register">
   </p>
 </form>
```

```
 <div id="create-account-wrap">
   <!-- <p>Not a member? <a href="#">Create Account</a><p> -->
 </div>
</div>
</body>
</html>
```

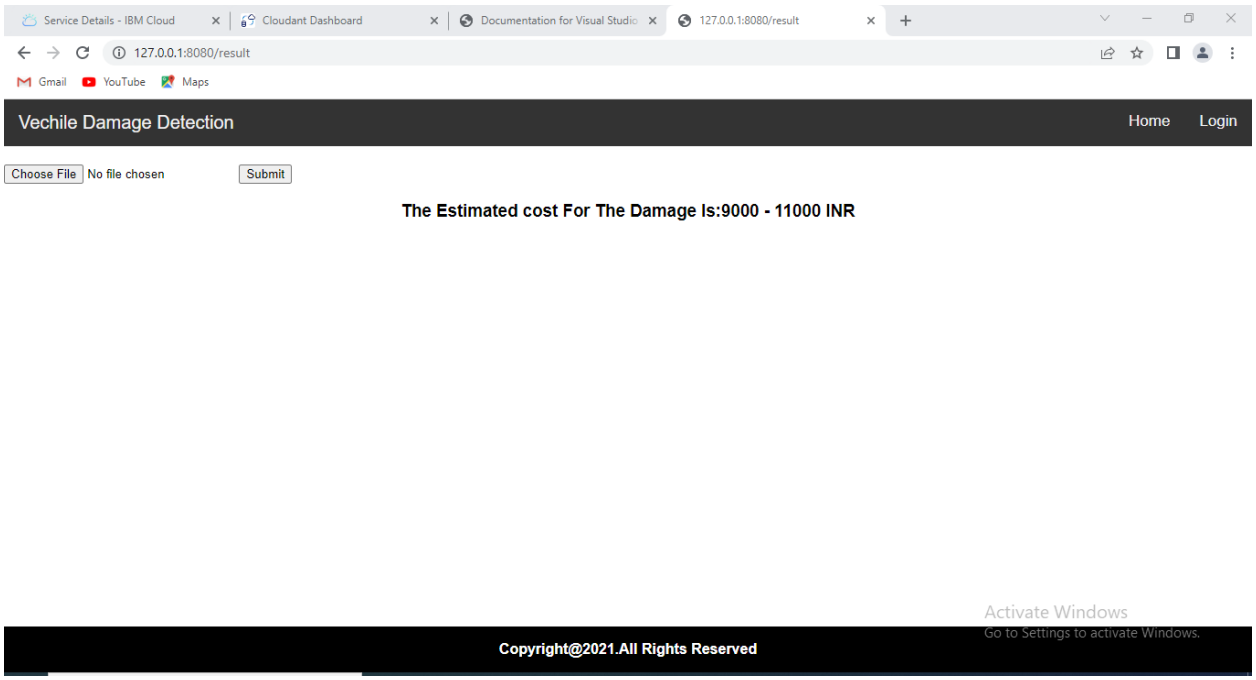


Insurance Prediction:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>

 <form id="" method="POST" action="/result" enctype="multipart/form-data">
 <div class="topnav">
       <a  id="logo" href="#"> Vehicle Damage Detection</a>
```

```
      <a href="{{ url_for('logout') }}">Logout</a>
      <a href="{{ url_for('index') }}">Home</a>
    </div>
  <div style="padding-top: 20px;">
   <input type="file" name="image" id=""/>
   <input type="submit"/>
  </div>
  <div>
   {% if value %}
     <h3 style="text-align: center ;">The Estimated cost For The Damage
Is: {{value}} </h3>
   {% endif %}
 </div>
 </form>
    <div class="footer">
      <p>Copyright@2021.All Rights Reserved</p>
    </div>
</body>
</html>
```



Python code:

```
#prediction
@app.route('/result',methods=["GET","POST"])
def result():
   if request.method=="POST":
       f=request.files['image']
       basepath=os.path.dirname(__file__) #getting the current path Le
where app.py #print("current path", basepath)
       filepath=os.path.join(basepath,'uploads',f.filename) #print("upload
folder is", filepath)
       f.save(filepath)
       img=image.load_img(filepath, target_size=(224,224))
       x=image.img_to_array(img)#ing to array x-np.expand dims(x,axis-e)
used for adding
       x=np.expand_dims(x,axis=0)
       img_data=preprocess_input(x)
       prediction1=np.argmax(model1.predict(img_data))
       prediction2=np.argmax(model2.predict(img_data))

       index1=['front','rear', 'side']
       index2=['minor', 'moderate', 'severe']

       result1=index1[prediction1]
       result2=index2[prediction2]

       if(result1=='front' and result2=='minor'):
           value="3000 - 5000 INR"
       elif(result1=='front' and result2=='moderate'):
           value="6000 - 8000 INR"
       elif(result1=='front' and result2=='severe'):
           value="9000 - 11000 INR"
       elif(result1=='rear' and result2=='minor'):
           value="4000 - 6000 INR"
       elif(result1=='rear' and result2=='moderate'):
           value="7000 - 9000 INR"
       elif(result1=='rear' and result2=='severe'):
           value="11000 - 13000 INR"
       elif(result1=='side' and result2=='minor'):
           value="6000 - 8000 INR"
```

```
    elif(result1=='side' and result2=='moderate'):
        value="9000 - 11000 INR"
    elif(result1=='side' and result2=='severe'):
        value="12000 - 15000 INR"
    else:
        value="16000 - 50000 INR"

    return render_template('prediction.html',value=value)
```

8.TESTING:

8.1 Test Cases:

	Test Scenarios
1	Verify user is able to see login page
2	Verify if the user is able to login to the application or not?
3	Verify users are able to navigate to the register page?
4	Verify user is able to upload images
5	Verify results

8.2 User Acceptance Testing Report:

1. Purpose of Document:

The purpose of this document is to briefly explain the test coverage and open issues of the Intelligent Vehicle Damage Assessment & Cost Estimator for Insurance Companies project at the time of the release to User Acceptance Testing (UAT).

2. 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	10	4	2	3	20

Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	18	35
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	1	0	0	1
Totals	24	14	13	26	77

3. Test Case Analysis:

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

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	50	0	0	50
Security	4	0	0	4
Outsource Shipping	3	0	0	3
Exception Reporting	8	0	0	8
Final Report Output	5	0	0	5
Version Control	2	0	0	2

8.3 Test Cases Report:

Testcases Report.xlsx

9. RESULTS:

9.1 Performance Metrics:

Performance Testing - Artificial Intelligence

10.ADVANTAGES & DISADVANTAGES:

ADVANTAGES:

- Easy to utilize claim process thanks to digitization
- Provide a thorough analysis of the damaged vehicle.
- Helps in the analysis of the damaged car and the payment process by the insurance company.

DISADVANTAGES:

- The manual method for submitting an insurance claim will take longer.
- The corporation acts improperly and currently doesn't make payments as a result of false accusations.
- Poor customer service.

11.CONCLUSION:

The concerns of car damage analysis and position and severity prediction will be dealt with in this research proposal using a neural network-based automobile detecting method. This project does a number of tasks in one go. The technique will undoubtedly help the insurance companies carry out much more complete and organized examinations of the vehicle damage. Simply providing a snapshot of the vehicle will allow the system to examine it, identify whether any damage is present, where it is located, and how bad it is.

12.FUTURE SCOPE:

We will need to use a large dataset and a variety of regularization techniques in our next work. If we have higher quality datasets that include the characteristics of a car (make, model, and year of production), location data, kind of damaged part, and repair cost, we can predict the cost of a broken automotive component more correctly and reliably. Together with a focus on the vehicle insurance sector, this study paves the way for future photo recognition efforts. By removing human bias, the study was able to accurately validate the existence of damage, its location, and its severity. By including the on-the-fly data augmentation methodologies, they can be further improved.

13.APPENDIX:

SOURCE CODE:

app.py

```
import numpy as np
import os
from flask import Flask, app, request, render_template
from tensorflow.keras import models
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from tensorflow.python.ops.gen_array_ops import concat
from tensorflow.keras.applications.inception_v3 import preprocess_input
import requests
from flask import Flask, request, render_template, redirect, url_for

# #Loading the model
from cloudant.client import Cloudant



# Authenticate using an IAM API key
client = Cloudant.iam('username','api key',connect=True)

# Create a database using an initialized client
my_database = client.create_database ('my_database')

model1=load_model('body.h5')
model2=load_model('level.h5')

app=Flask(__name__)

#default home page or route
@app.route('/')
def index():
   return render_template('index.html')

#register page
@app.route('/register')
def register():
```

```
    return render_template('register.html')

@app.route('/afterreg',methods=['POST'])
def afterreg():
    x=[x for x in request.form.values()]
    print(x)
    data={
        '_id':x[1],
        'name':x[0],
        'psw':x[2]
    }
    print(data)
    query={'_id':{'$eq':data['_id']}}
    docs=my_database.get_query_result(query)
    print(docs)

    print(len(docs.all()))

    if(len(docs.all())==0):
        url=my_database.create_document(data)
             return  render_template('login.html',pred="Registration  successful,
Please login your details")
    else:
          return  render_template('register.html',pred="You  re  already  member,
Please login using r details")


#login page
@app.route('/login')
def login():
    return render_template('login.html')

@app.route('/afterlogin',methods=['POST'])
def afterlogin():
    user=request.form['_id']
    password=request.form['psw']
    print(user,password)
    query={'_id':{'$eq':user}}

    docs=my_database.get_query_result(query)
    print(docs)
```

```
   print(len(docs.all()))

   # if(len(docs.all())==0):
   #     return render_template('login.html',pred="The username not found")
   # else:
   if(user==docs[0][0]['_id'] and password==docs[0][0]['psw']):
       return redirect(url_for('prediction'))
   else:
       return render_template('error.html')

#prediction page
@app.route('/prediction')
def prediction():
   return render_template('prediction.html')
#logout page
@app.route('/logout')
def logout():
   return render_template('logout.html')

#prediction
@app.route('/result',methods=["GET","POST"])
def result():
   if request.method=="POST":
       f=request.files['image']
         basepath=os.path.dirname(__file__) #getting the current path Le where
app.py #print("current path", basepath)
            filepath=os.path.join(basepath,'uploads',f.filename)  #print("upload
folder is", filepath)
       f.save(filepath)
       img=image.load_img(filepath, target_size=(224,224))
        x=image.img_to_array(img)#ing to array x-np.expand dims(x,axis-e) used
for adding
       x=np.expand_dims(x,axis=0)
       img_data=preprocess_input(x)
       prediction1=np.argmax(model1.predict(img_data))
       prediction2=np.argmax(model2.predict(img_data))

       index1=['front','rear', 'side']
       index2=['minor', 'moderate', 'severe']

       result1=index1[prediction1]
       result2=index2[prediction2]
```

```
        if(result1=='front' and result2=='minor'):
            value="3000 - 5000 INR"
        elif(result1=='front' and result2=='moderate'):
            value="6000 - 8000 INR"
        elif(result1=='front' and result2=='severe'):
            value="9000 - 11000 INR"
        elif(result1=='rear' and result2=='minor'):
            value="4000 - 6000 INR"
        elif(result1=='rear' and result2=='moderate'):
            value="7000 - 9000 INR"
        elif(result1=='rear' and result2=='severe'):
            value="11000 - 13000 INR"
        elif(result1=='side' and result2=='minor'):
            value="6000 - 8000 INR"
        elif(result1=='side' and result2=='moderate'):
            value="9000 - 11000 INR"
        elif(result1=='side' and result2=='severe'):
            value="12000 - 15000 INR"
        else:
            value="16000 - 50000 INR"

        return render_template('prediction.html',value=value)


#run app
if __name__ == "__main__":
   app.run(debug=True,port=8080)
```

register.html

```
<!DOCTYPE html>
<html lang="en" >
<head>
 <meta charset="UTF-8">
 <title>Register</title>
 <!-- <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/normalize/5.0.0/normalize.min.css
">
<link rel="stylesheet" href="E:\IBM\static\styles\Register.css"> -->
```

```
</head>
<body>
 <div class="topnav">
   <a  id="logo" href=""> Vehicle Damage Detection</a>
   <a href="{{ url_for('register') }}">Register</a>
   <a href="{{ url_for('login') }}">Login</a>
   <a href="{{ url_for('index') }}">Home</a>
 </div>
<div id="login-form-wrap">
 <h2>Register</h2>
 <form id="login-form" method="POST" action="/afterreg">
   <p>
       <input type="text" id="text" name="name" placeholder="Username"
required><i class="validation"><span></span><span></span></i>
   </p>
   <p>
       <input type="email" id="email" name="email" placeholder="Email Address"
required><i class="validation"><span></span><span></span></i>
   </p>
   <p>
       <input type="password" id="password" name="psw" placeholder="Password"
required=""><i class="validation"><span></span><span></span></i>
   </p>
   <p>
   <input type="submit" id="login" value="Register">
   </p>
 </form>
 <div id="create-account-wrap">
   <!-- <p>Not a member? <a href="#">Create Account</a><p> -->
 </div>
</div>
</body>
</html>
```

login.html:

```
<!DOCTYPE html>
<html lang="en" >
<head>
 <meta charset="UTF-8">
 <title>Login</title>
```

```
 <!-- <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/normalize/5.0.0/normalize.min.css
">
<link rel="stylesheet" href="E:\html\login.css"> -->
</head>
<body>
 <div class="topnav">
   <a  id="logo" href="#">Vehicle Damage Detection</a>
   <a href="{{ url_for('register') }}">Register</a>
   <a href="{{ url_for('login') }}">Login</a>
   <a href="{{ url_for('index') }}">Home</a>
 </div>
   <div id="login-form-wrap">
     <h2>Login</h2>
     <form id="login-form" method="POST" action="/afterlogin">
       <p>
           <input type="email" id="email" name="_id" placeholder="Email
Address" required><i class="validation"><span></span><span></span></i>
       </p>
       <p>
           <input type="password" id="password" name="psw"
placeholder="Password" required><i
class="validation"><span></span><span></span></i>
       </p>
       <p>
       <input type="submit" id="login" value="Login">
       </p>
     </form>
     <div id="create-account-wrap">
   </div>
 </body>
</html>
```

prediction.html:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {
 margin: 0;
```

```
 font-family: Arial, Helvetica, sans-serif;
}

.topnav {
 overflow: hidden;
 background-color: #333;
}

.topnav a {
 float: right;
 color: #f2f2f2;
 text-align: center;
 padding: 14px 16px;
 text-decoration: none;
 font-size: 17px;
}

#logo{
 float: left;
 font-size: 20px;
 text-decoration: none;
}

.footer {
 position: fixed;
 left: 0;
 bottom: 0;
 width: 100%;
 font-weight: bold;
 background-color: black;
 color: white;
 text-align: center;
}
</style>
</head>
<body>

 <form id="" method="POST" action="/result" enctype="multipart/form-data">
 <div class="topnav">
       <a  id="logo" href="#">Vehicle Damage Detection</a>
       <a href="{{ url_for('logout') }}">Logout</a>
       <a href="{{ url_for('index') }}">Home</a>
```

```
    </div>
  <div style="padding-top: 20px;">
   <input type="file" name="image" id=""/>
   <input type="submit"/>
  </div>
  <div>
   {% if value %}
     <h3 style="text-align: center ;">The Estimated cost For The Damage Is:
{{value}} </h3>
   {% endif %}
 </div>
 </form>
     <div class="footer">
       <p>Copyright@2021.All Rights Reserved</p>
     </div>
</body>
</html>
```

index.html

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {
 margin: 0;
 font-family: Arial, Helvetica, sans-serif;
}

.topnav {
 overflow: hidden;
 background-color: #333;
}

.topnav a {
 float: right;
 color: #f2f2f2;
 text-align: center;
 padding: 14px 16px;
 text-decoration: none;
 font-size: 17px;
```

```
}

#logo{
 float: left;
 font-size: 20px;
 text-decoration: none;
}


.content{
 text-align: center;
}
.content-button{
 margin-left: 50%;
 background-color: #333;
 padding: 5px 10px;
 color: white;
}

.para{
 padding-top: 25px;
 text-align: center;
 font-size: 22px;
}

</style>
</head>
<body>

<div class="topnav">
 <a  id="logo" href="#">Vehicle Damage Detection</a>
 <a href="{{ url_for('register') }}">Register</a>
 <a href="{{ url_for('login') }}">Login</a>
 <a href="{{ url_for('index') }}">Home</a>
</div>


<div>
 <h2 class="content">About Project</h2>
 <div class="para">
   <p class="contentPara">Vehicle detection is used to reduce claims leakage
during insurance processing.</p>
```

```
   <p class="contentPara">Visual inception and validation are usually done.As it takes a long time, because a person needs to come and inspect the damage.</p>
   <p class="contentPara">Here we are trying to automate the procedure. Using this automation, we can avoid time conception for the insurance claim process</p>
 </div>
</div>

</body>
</html>
```

logout.html

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<style>
body {
 margin: 0;
 font-family: Arial, Helvetica, sans-serif;
}

.topnav {
 overflow: hidden;
 background-color: #333;
}

.topnav a {
 float: right;
 color: #f2f2f2;
 text-align: center;
 padding: 14px 16px;
 text-decoration: none;
 font-size: 17px;
}

#logo{
 float: left;
 font-size: 20px;
 text-decoration: none;
```

```
}


.content{
 text-align: center;
 padding-left: 37px;
}
.content-button{
 margin-left: 50%;
 background-color: #333;
 padding: 5px 10px;
 color: white;
 text-decoration: none;
}
</style>
</head>
<body>

<div class="topnav">
 <a  id="logo" href="#">Vehicle Damage Detection</a>
 <a href="{{ url_for('register') }}">Register</a>
 <a href="{{ url_for('login') }}">Login</a>
 <a href="{{ url_for('index') }}">Home</a>
</div>


<div>
 <h2 class="content">Successfully Logged out</h2>
 <p class="content">Login for more information</p>
 <a class="content-button" href="{{ url_for('login') }}">Login</a>
</div>

</body>
</html>
```

GitHub Link: https://github.com/IBM-EPBL/IBM-Project-1062-1658338087

Project Demo Link: https://drive.google.com/file/d/1eoa7_1q5_ygTITQZOt4Bgo2P5f5MZwL4/view?usp=sharing