Project On

# Intelligent Vehicle Damage Assessment & Cost Estimator for Insurance Companies

powered By IBM India


Submitted By

**MURALITHARA.R**

**POORNIMA.A**

**SANJAI KUMAR.S**

**SURYA.N**


**Project ID: PNT2022TMID09778**

of


**Department of INFORMATION TECHNOLOGY**
**ST.JOSEPH'S COLLEGE OF ENGINEERING**
**OMR, Chennai - 600119**

College Mentor: S L SWARNA
Industrial Mentor: SWATHI

# INDEX

# 1. INTRODUCTION

## Project Overview

The project "Intelligent Vechile Damage Assessment and Cost Estimator for Insurance Companies" is a responsive web application powered by aritifical Intelligence and IBM Watson Cloud. Deep Learning model is trained with the various damaged car images in various views and the VGG16 from the TensorFlow library is used for the better Deep Learning model architecture. An attractive front end can be developed using HTML and CSS. The pages such as Index.html , login.html, logout.html, register.html and prediction.html are created and embedded with the IBM cloud databse using python framework called flask. The web application takes the image input and estimate the cost for the insurance companies based on the damages in the car.

## Purpose

The project is based on the domain of Artificial Intelligence and powered by the IBM watson cloud. A responsive web application can be developed using the HTML and CSS which is connected to waston cloud. In the cloud, a database service by availing the service Instance of the IBM cloud and the database API key is collected and connected with the front-end using flash which is an python framework for desiging the backend. Pages such as index.html, login.html, logout.html and prediction.html are used to interact with the web application. The user can register and the data of the user is saved in the databse of the IBM cloud, during the time of login, the login ID is compared with the ID in the databse and allow the user to the next page. The Deep Learning model is build using the VGG16 which is present in the keras library and the model is trained with the images of mulitple car with various level cum types of damages. The model is deployed in the back-end using the flask and the prediction.html page is setted to collect the image from the user. The prediction algorithm is used treat the image and estimated the cost for the user. The project is based on the various components which helps to handle the back - end and Front - end. Then front - end is build using html and css which is connectedback - end which is build using the python and IBM cloud. The project is powered by the IBM Watson cloud and is based in the artificial intelligence field. With the use of HTML and CSS and the Waston Cloud, a responsive web application may be created. The database API key is gathered and connected with the front-end using flash, which is a python framework for designing the backend.

# 2.LITERATURE SURVEY

| TITLE | AUTHOR | YEAR | DESCRIPTION | ADVANTAGES | DISADVANTAGES | METHODOLOGY |
|---|---|---|---|---|---|---|
| The use of telematics data in vehicle insurance | • K.Korishchenko<br>• I. Stankevich<br>• N. Pilnik<br>• D. Petrova | 2019 | This paper introduces an approach to telematic devices data application in automotive insurance. we conduct a comparative analysis of different types of devices that collect information on vehicle utilization and driving style of its driver | Better Customer Service, Improved Risk Management, Incresed Client Base. | It is easy to track the user because of sending privacy information to the system. Installing telematics is expensive | Telematics data such as GNSS positioning and communicate via server (GSM/GPRS) |
| we can use well-organized deep knowledge-based construc | • Jihabqaddour<br>• Syeda Ayesha siddiqa | 2022 | This paper shows deep learning algorithms have been | Quickly accessing claims, verifying documents, enhancing customer | Incapable of multitasking, hardware dependence, deep learning models will perform well | We get attainable databases from damaged automobile vehicles. focus on |

| | | | | | | |
|---|---|---|---|---|---|---|
| tions for detecting, localizing, and classification vehicle damage using enhanced Mask R-CNN method which integrates deep learning, instance segmentation. | | | utilized to clarify such issues, mitigate their pessimistic implications, and automate this process to save time and money. In this context, we offer brutalize as well as well-organized deep knowledge-based constructions for detecting, localizing, and classification vehicle damage using enhanced Mask R-CNN method | experience and detecting fraud | when their complexity is appropriate to complexity of the data. | augmentation of data to enlarge syntheticall y and alter the data set to relax its tolerance and improve its performance to the problem of overfitting at the time of training as we already work with a limited set of data. |
| Research on Vehicle Appearance Damage Recognition Based | • Qianqian Zhu[1],<br>• Wei Hu[2],<br>• Yingnan Liu[1] and Zihao Zhao[1] | 2021 | This paper shows how vehicle appearance damage recognizes the algorithm based on deep | qualitative improvement compared with the traditional car insurance claim settlement business. good robustness | it is difficult to define the obvious scope of a vehicle damage. | Damage Recognition Model.The model includes the backbone network for feature extraction, the RPN generated by candidate |

| | | | | | | |
|---|---|---|---|---|---|---|
| on Deep Learning | | | learning and evaluation model. | s and high accuracy. | | regions, ROIAlign and the headnetwork. Evaluation Method Based on Component-assisted. Evaluation Method Based on Component-assisted. |
| Damaged Vechile parts Recognition Using Capsule Neural Network | KundjanasithThonglek NorawitUrailertprasertPatcharaPattiyathaneeChantanaChantrapornchai | 2022 | This paper shows about Damgedvechile part recgonition using capsule neural network.a damaged vehicle part detection platform, called Intelligent Vehicle Accident Analysis (IVAA) which provides artificial intelligence as a service (AIaaS), is proposed. | damage localization and damage classification. The accuracy of the damage localization is 93.28% and the accuracy of the damage classification is 98.47%, respectively. | error-prone, time-consuming, and requires man-hour workers. | The system helps automatically assess vehicle parts' damage and severity level. An insurance company can utilize our service to speed up the claiming process. IVAA is built on the docker image which allows the system to be scaled depending on the workload efficiently. Capsule neural network (CapsNet) is applied for damage recognition including |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | | | | | two phrases: damage localization and damage classification. The accuracy of the damage localization is 93.28% and the accuracy of the damage classification is 98.47%, respectively. |
| Accurate Damage Dimension Estimation in AI Driven Vehicle Inspection System | AdritaBarari<br>N.V.S Abhilash<br>Payanshi Jain<br>Ankit Sati<br>Karthik Sai Datta<br>Chirag Jain | 2020 | In this paper, they present a near real-time end-to-end solution which yields accurate damage detection and propose approaches for providing dimensions of the damages for accurate repair cost estimates for the vehicle. | The damage assessment involves granular part detection, damage localization and classification into different damage types such as dent, scratch, crush, tear, etc. | Physical dimension estimation of the damages is required for computing cost to the customer. | Incorporates an ensemble of computer vision algorithms to calculate dimensions, generating bounding boxes, and consolidating the damage predictions to provide an overall damage estimate from multiple images captured by vehicle owner. |

| | | | | | | |
|---|---|---|---|---|---|---|
| Car Damage Detection and Cost Evaluation Using MASK R-CNN | J. D. DorathiJayaseeli<br>GreetaKavithaJayaraj<br>MehaaKanakarajan<br>D. Malathi | 2021 | In this paper, they propose employing convolution neural networks to build a Mask R-CNN model that can detect the area of damage on a car. The dataset used consists of images of damaged vehicles with a single class named scratch. | Car dealers can eliminate the manual process of damage assessment and the labor cost accompanied by it. Accuracy and transparency in pricing cars and their potential repairs will be made more prevalent Fraudulent vehicle insurance claims can also be diminished | Vehicles have very reflective metallic bodies the photographs taken in such an uncontrolled environment can be expected to have a considerable amount of inter object reflection. Therefore, the application of standard computer vision techniques in this context is a very challenging task. | Detecting the damage on a car using image-based processing method with enormous scope for automation. |

| | | | | | | |
|---|---|---|---|---|---|---|
| Research on Intelligent Vehicle Damage Assessment System Based on Computer Vision | Zhu Qianqian<br>GuoWeiming<br>Shen Ying<br>Zhao Zihao | 2020 | In this paper, based on the demand of automobile insurance claims and intelligent transportation, combined with abundant basic data and advanced machine vision algorithm, an intelligent damage determination system of 'Artificial Intelligence + Vehicle Insurance' is constructed. This paper first introduces the functions of the intelligent damage assessment system | It assists insurance companies to achieve rapid and accurate pricing in the process of fixing losses and claims.<br><br>It realizes the fraud recognition in the whole process of damage determination and can effectively control the cost expenditure of insurance companies. | The photography of vehicle damage,it is necessary to shoot the vehicle damage head-on so that the damage location is as far as possible in the center of the picture. The shooting distance is about 1 meter, and it is suitable to shoot clearly | Detecting the damage on a car using Computer Vision method |

# 3.IDEATION & PROPOSED SOLUTION

Project team shall fill the following information in proposed solution template.

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | In this project , we use car images for detecting the damages and calculated the associated cost |
| 2. | Idea / Solution description | Computer Vision and Deep Learning techniques to accurately classify vehicle damage to facilitate claims triage by training convolution neural networks |
| 3. | Novelty / Uniqueness | Damages on the car is detected using computer vision and the cost will be calculated for the associated damages |
| 4. | Social Impact / Customer Satisfaction | This project will help the people to know the cost for the damage and know the various different damages that happened to the car |
| 5. | Business Model (Revenue Model) | This product can be sold to car insurance companies |
| 6. | Scalability of the Solution | CNNs are primarily used for image classification and recognition. The specialty of a CNN is its convolutional ability. The potential for further uses of CNNs is limitless and needs to be explored and pushed to further boundaries to discover all that can be achieved by this complex machinery. |

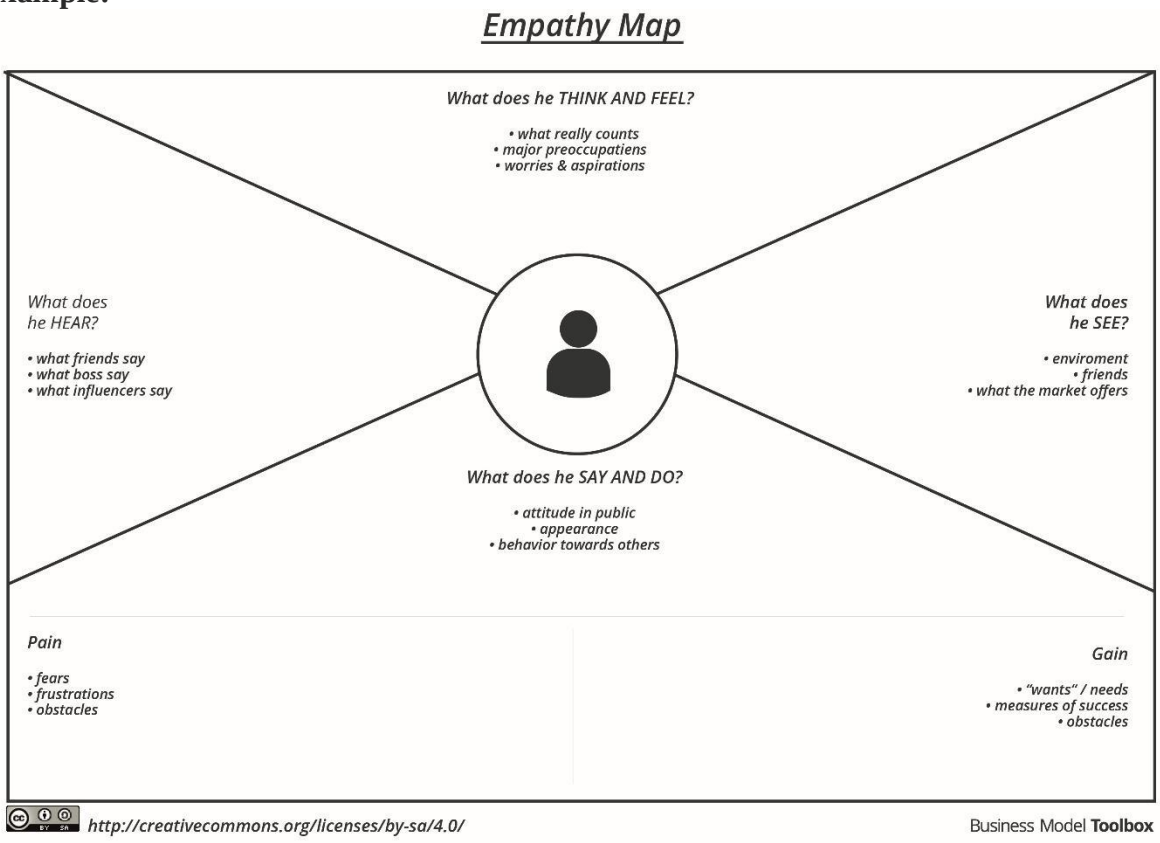
## Empathy Map

1. An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes.

2. It is a useful tool to helps teams better understand their users.

3. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user's perspective along with his or her goals and challenges.

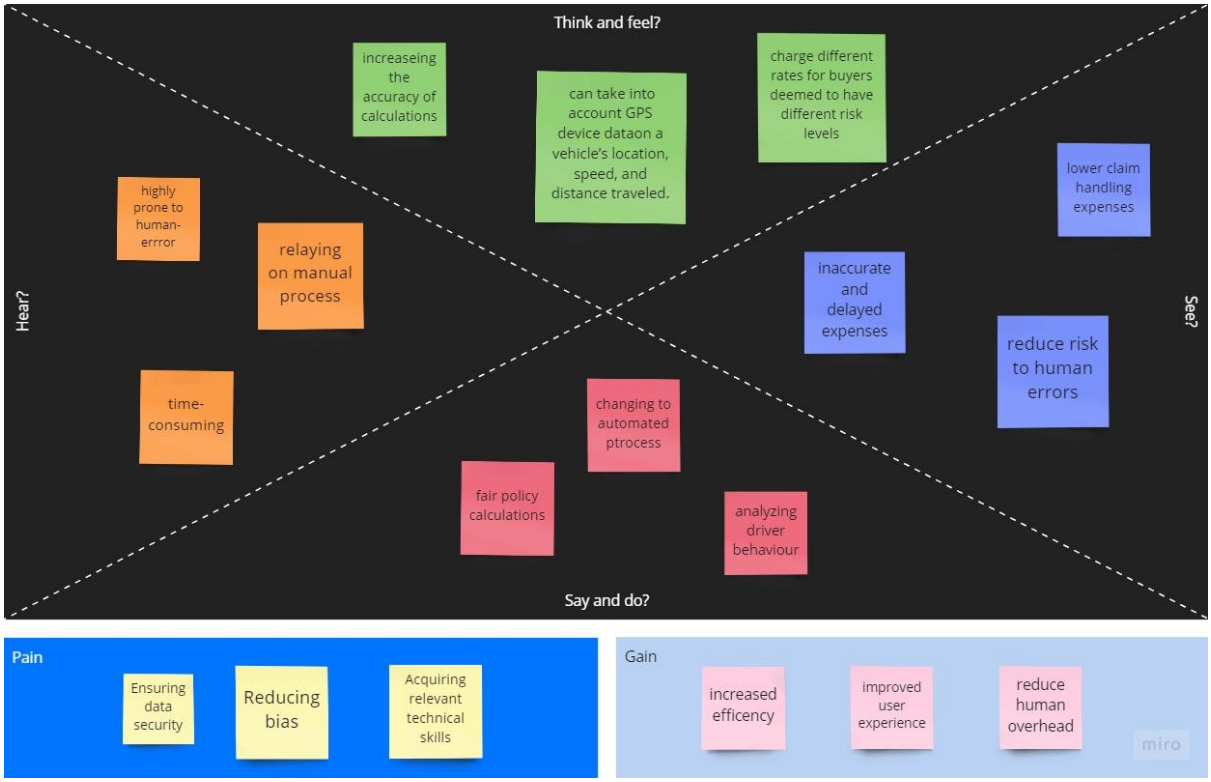
**Example:**




Reference: https://www.mural.co/templates/empathy-map-canvas

**Example:**

**Intelligent Vehicle Damage Assessment & Cost Estimator for Insurance Companies**



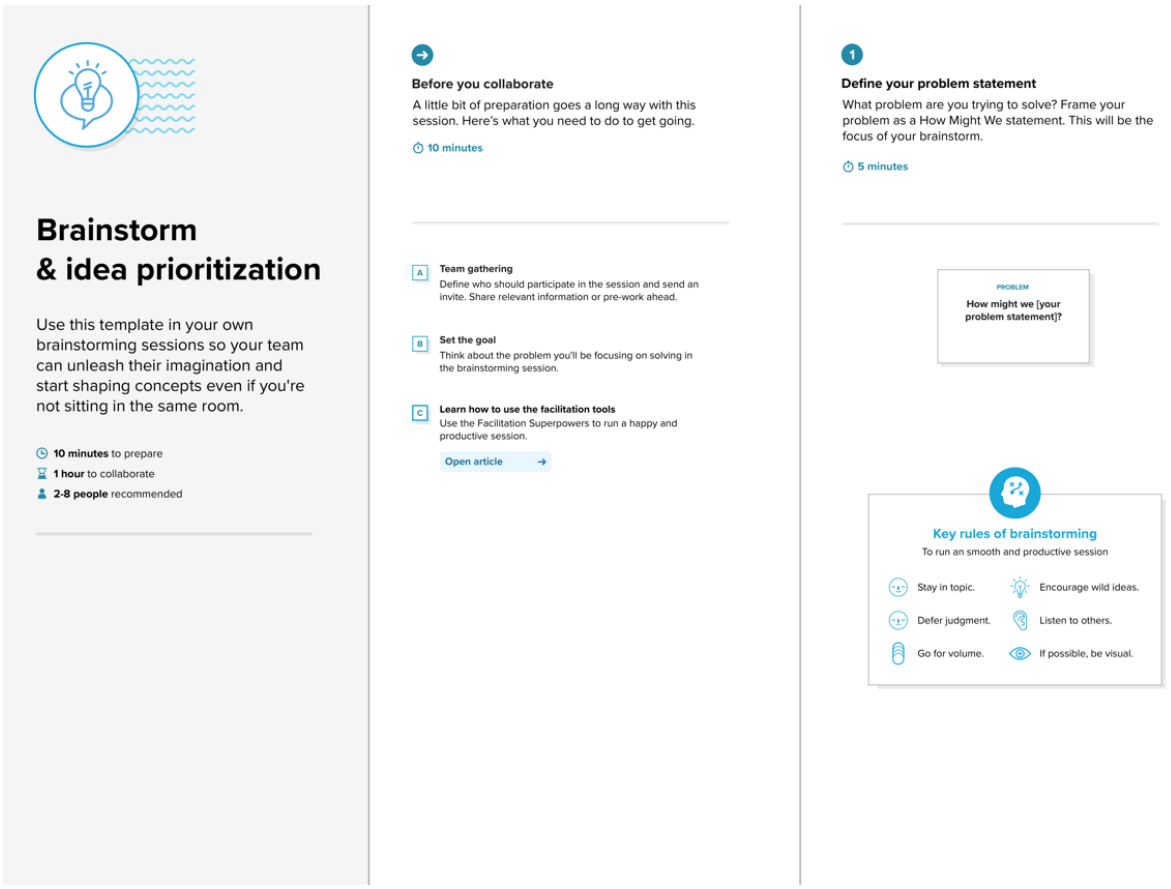

**Brainstorm & Idea Prioritization Template:**

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

Reference: https://www.mural.co/templates/empathy-map-canvas

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




**Step-2: Brainstorm, Idea Listing and Grouping**

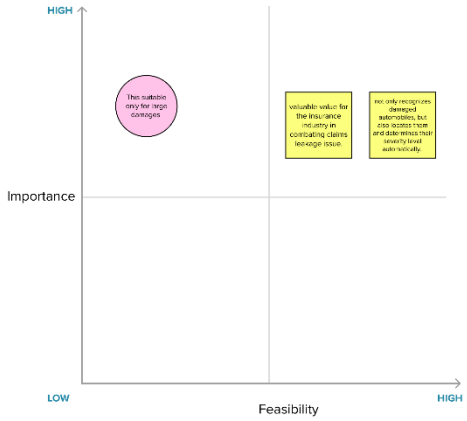
# Step-3: Idea Prioritization

3

**Sort, share, and discuss**

Sort your ideas into the 2x2 template(s) and look for new insights on each idea as the grid takes shape. Play with a few more re-sorts along additional criteria sets as a team. Review and discuss.

30 minutes



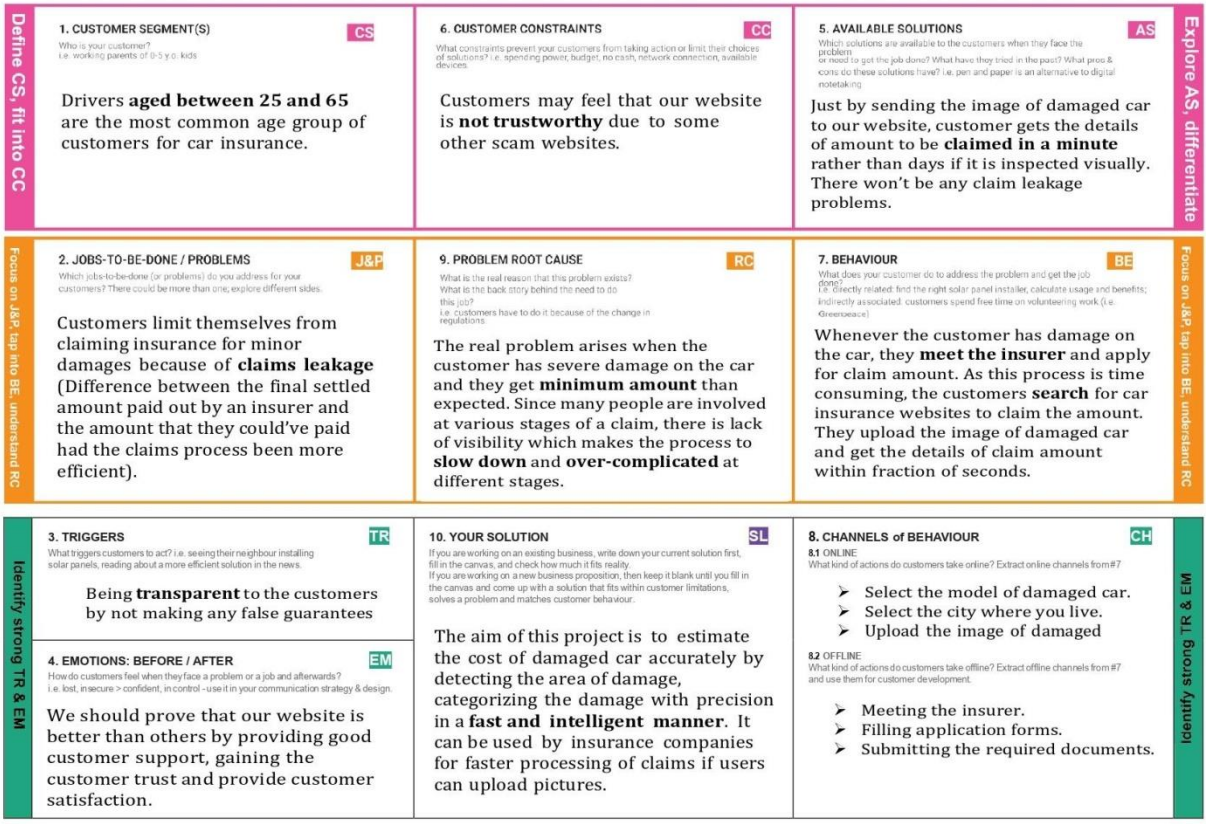
**Píoposed Solution**

Project team shall fill the following information in proposed solution template.

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (Problem to be solved) | In this project , we use car images for detecting the damages and calculated the associated cost |
| 2. | Idea / Solution description | Computer Vision and Deep Learning techniques to accurately classify vehicle damage to facilitate claims triage by training convolution neural networks |
| 3. | Novelty / Uniqueness | Damages on the car is detected using computer vision and the cost will be calculated for the associated damages |
| 4. | Social Impact / Customer Satisfaction | This project will help the people to know the cost for the damage and know the various different damages that happened to the car |
| 5. | Business Model (Revenue Model) | This product can be sold to car insurance companies |
| 6. | Scalability of the Solution | CNNs are primarily used for image classification and recognition. The specialty of a CNN is its convolutional ability. The potential for further uses of CNNs is limitless and needs to be explored and pushed to further boundaries to discover all that can be achieved by this complex machinery. |

# Píoposed Solution Ïit

**Project Title:** Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies  **Project Design Phase-I - Problem Solution Fit**  **Team ID: PNT2022TMID00513**



# 4.REQUIREMENT ANALYSIS

## 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<br>Registration through Gmail |
| FR-2 | **User Confirmation** | Confirmation via Email<br>Confirmation via OTP |
| FR-3 | **User details** | Users are required to register their personal details. like name, age, date of birth, driving license, car number etc. |
| FR-4 | **User requirements** | The user simply inputs vehicle damage images. The software will instantly generate an accurate reading of the based on the image detection analysis in a readable format familiar to the customer. It compares the information already given and states the defect percentage and cost in that vehicle damage image . |

## NON FUNCTIONAL REQUIREMENTS

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

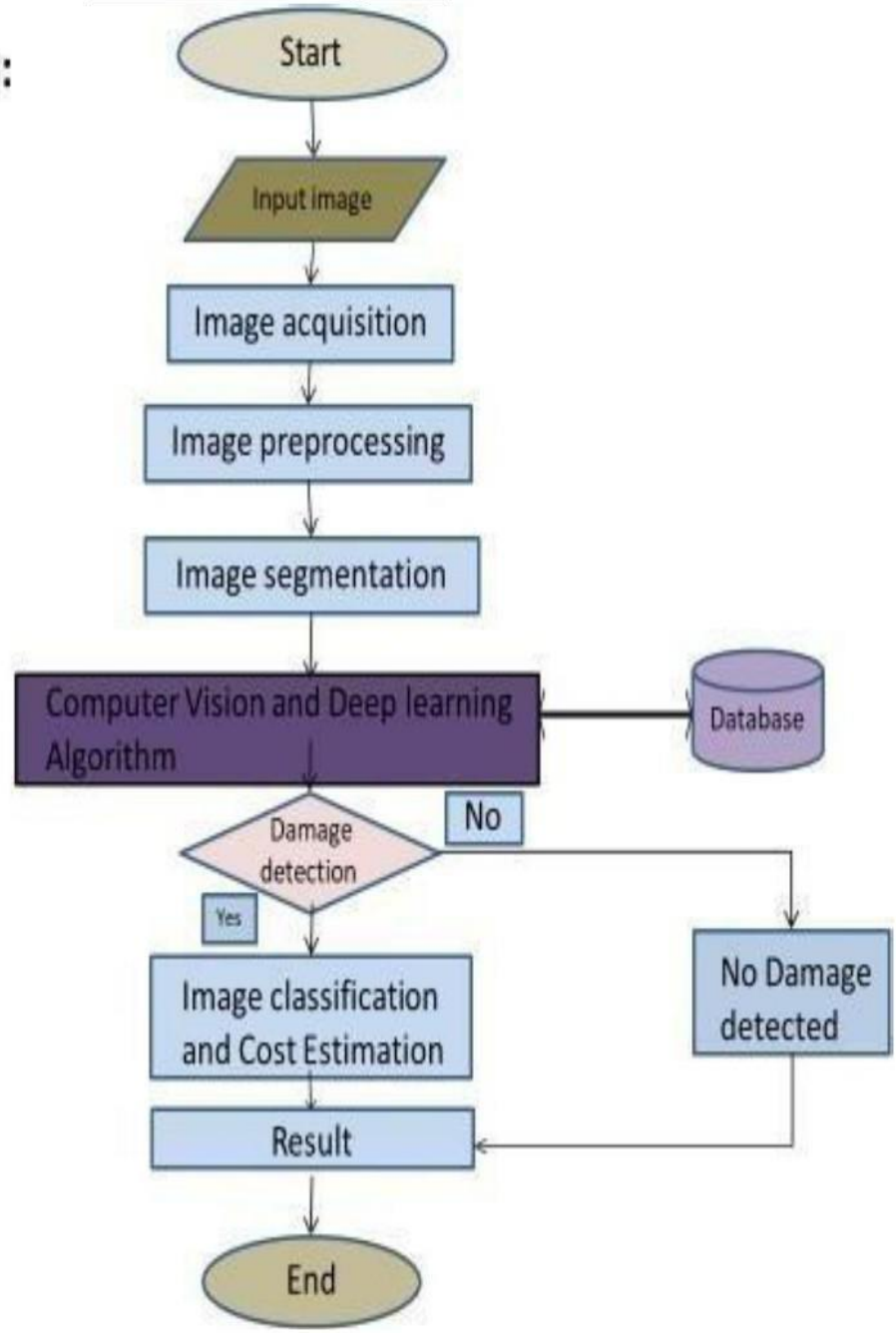
| FR No. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | **Usability** | More efficient for the frequent users. users can easily understand what the application does and feel satisfied with the system. |
| NFR-2 | **Security** | • AI powered vehicle damage assessment and cost estimator for insurance company should contain more security in which our data which entered or maintained should be more security.<br>• With the help of the username and password it provides more security in which it can access more securable and the data are private |
| NFR-3 | **Reliability** | This application must perform without failure in 90 percentage of use cases during a month. It is morereliable. |
| NFR-4 | **Performance** | This application supporting 1,050 users per hour must provide 5 seconds or less response time in a desktop browser, including the rendering of text and images, over an LTE connection. The performance of this application is effective and efficient. |
| NFR-5 | **Availability** | The web dashboard must be available to user's 99.9 percent of the time every month during business hours EST. Users can access any time and any where. |
| NFR-6 | **Scalability** | The application must be scalable enough to support 10,000 visits at the same time while maintaining optimal performance and efficient to retrieve image in large scale thus improving scalability. |

## 5.PROJECT DESIGN

## DATA FLOW DIAGRAM

**Data Flow Diagram**

Data Flow Diagram:



# SOLUTION AND TECHNICAL ARCHITECTURE

## Solution Architecture:

To automate such a system, the easiest method would be to build a Convolution Neural Network model capable of accepting images from the user and determining the location and severity of the damage. The model is required to pass through multiple checks that would first ensure that given image is that of a car and then to ensure that it is in fact damaged. These are the gate checks before the analysis begins. Once all the gate checks have been validated, the damage check will commence. The model will predict the location of the damage as in front, side or rear, and the severity of such a damage as in minor, moderate or severe.

The model accepts an input image from the user and processes it across 4 stages:

1. Validates that given image is of a car.
2. Validates that the car is damaged.
3. Finds location of damage as front, rear or side
4. Determines severity of damage as minor, moderate or severe

The model can also further be imporved to:

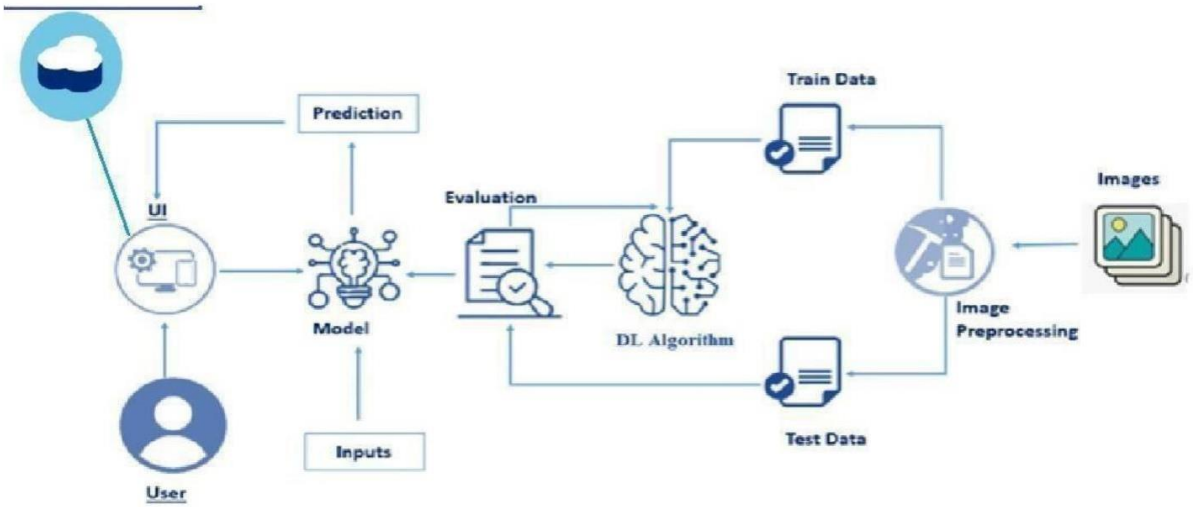
1. Obtain a cost estimate
2. Send assessment to insurance carrier
3. Print documentation

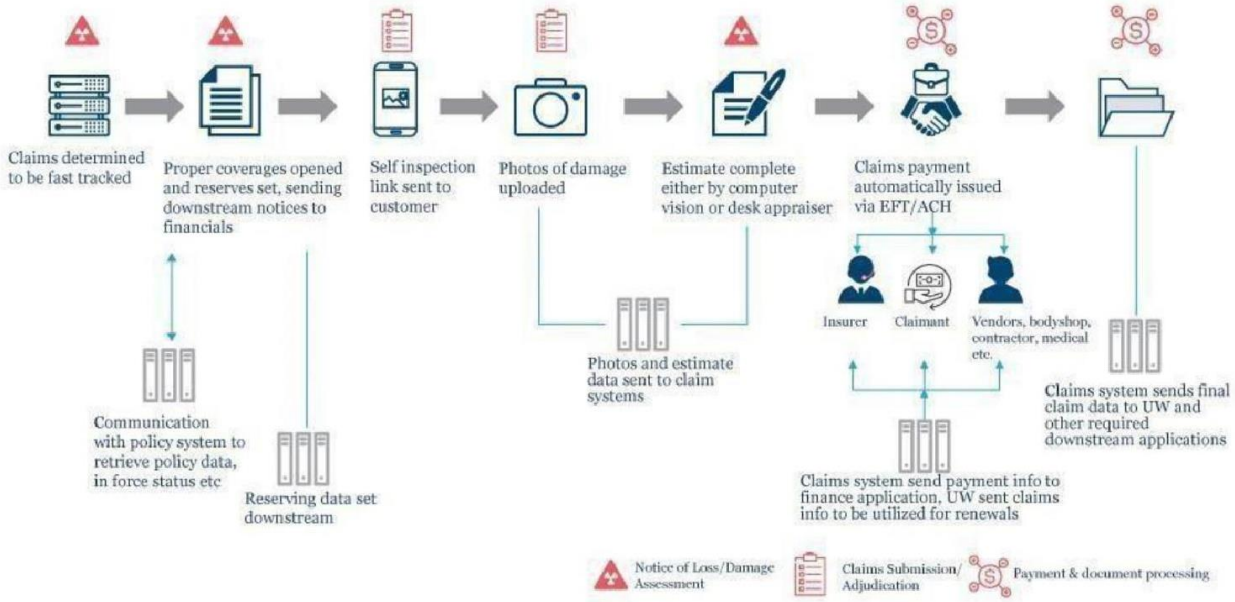
**PROCEDURE:**

- Data Collection.
- Collect the dataset or create the dataset
- Data Pre-processing.
- Import the Image Data Generator library
- Configure Image Data Generator class
- Apply Image Data Generator functionality to Train set and Test set
- Model Building
- Import the model building Libraries
- Initializing the model
- Adding Input Layer
- Adding Hidden Layer
- Adding Output Layer

- Configure the Learning Process
- Training and testing the model
- Save the Model
- Application Building
- Create an HTML file
- Build Python Code

**Technology Architecture:**



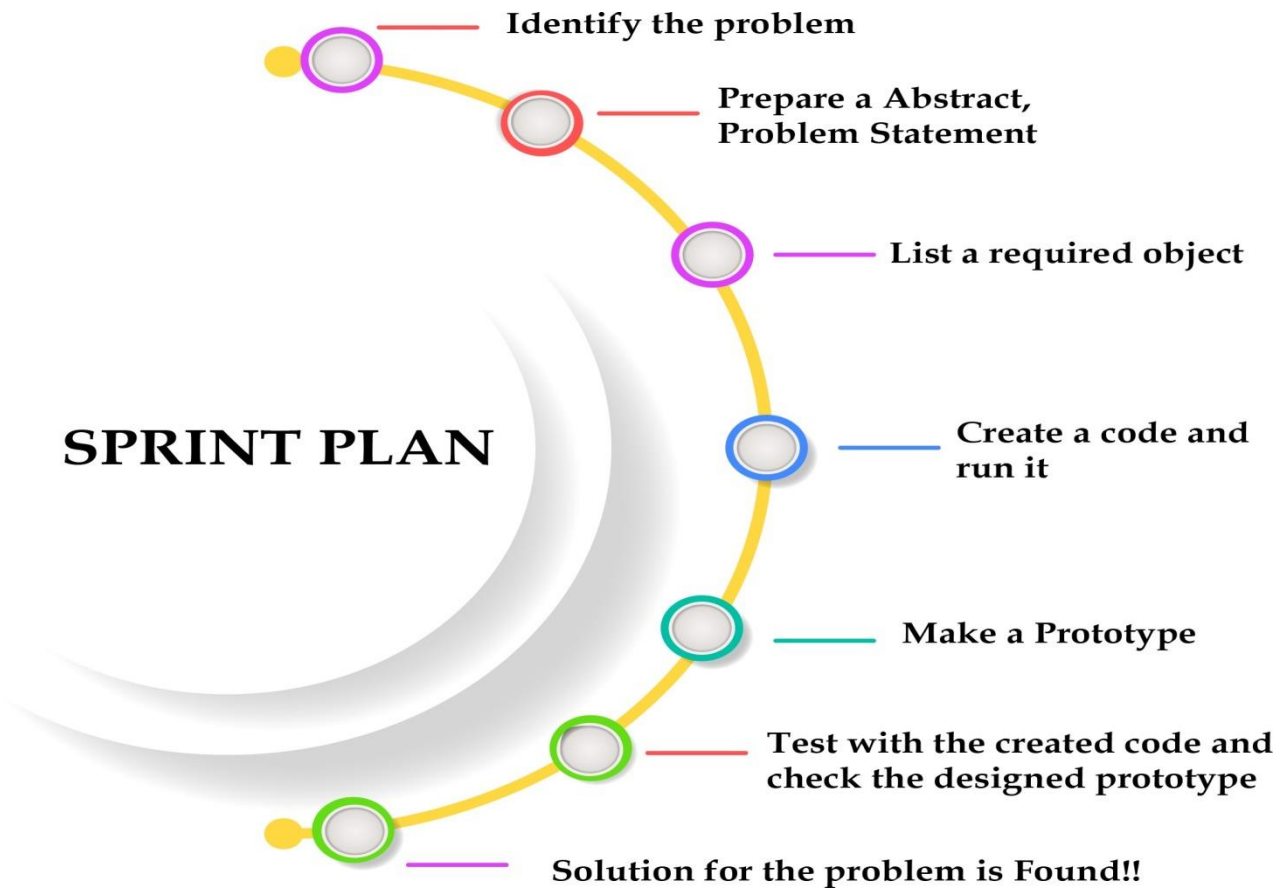



# USER STORIES

| Use Type | Functional Requirement (Epic) | User Story Number | User Story / Task | Acceptance criteria | Priority | Release |
|---|---|---|---|---|---|---|
| Customer (Mobile user) | Registration | USN-1 | As a user, I can register for the application by entering my email, password, and confirming my password. | I can access my account / dashboard by entering valid credentials | High | Sprint-1 |
| Customer Details | Login | USN-2 | As a user, I will receive confirmation email once I have registered for the application | I can receive confirmation email & click confirm | High | Sprint-1 |
| Customer Uses | Dashboard | USN-3 | As a user, I can register for the application through Facebook | I can register & access the dashboard with Facebook Login | Low | Sprint-4 |
| Customer Options | Details about insurance companies | USN-4 | As a user, I can register for the application through Gmail | I can register & access the dashboard with Facebook Gmail | Medium | Sprint-1 |
| Customer usage | Login | USN-5 | As a user, I can log into the application by entering email & password | I can log in and view my dashboard at my demand on any time | High | Sprint-1 |
| Customer needs to do | Dashboard | USN-6 | As a user I must capture images of my vehicle and upload it into the web portal | I can capture the entire vehicle and upload | High | Sprint-2 |
| Customer (Web user) | Details about estimated cost based on damage | USN-7 | As a user I must receive a detailed report of the damages present in the vehicle and the cost estimated | I can get the estimated insurance cost | High | Sprint-3 |
| Customer Care Executive | Details about Estimated cost Based on damage | USN-8 | As a user, I need to get support from developers in case of queries and failure of service provided | I can have smooth user experiences and all the issues raised is sorted | Medium | Sprint-4 |
| Administrator | Details about Estimated cost Based on damage | USN-9 | We need to satisfy the customer needs in an efficient way and make sure any sort of errors are fixed | I can finish the work without any problems | High | Sprint-4 |

# 6.PROJECT PLANNING & SCHEDULING

## SPRINT PLANNING AND ESTIMATION



## Product Backlog, Sprint Schedule, and Estimation:

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | Registration | USN - 1 | As a user, I can register for the application by entering my details of name, email, cars etc. verifying my Gmail account and creating new account with password | 7 | HIGH | TM-1,4 |
| Sprint-1 | Login | USN -2 | As a user, entering my email, and password, and confirming my password, I can login to myaccount. | 7 | HIGH | TM-1,4 |
| Sprint-1 | Dashboard | USN-3 | As a user, I can clearly see data, point, graphs, charts and trends of my previous activity and global activity related to my views | 2 | LOW | TM-1,4 |
| Sprint-2 | Details about insurance company | USN-4 | As a user, I can register for the application through Gmailand account id. | 8 | MEDIUM | `TM-2,3 |
| Sprint-1 | repeated logins and logout | USN-5 | As a user, I can log in and view my dashboard at my demand on any time | 4 | HIGH | TM-1,4 |
| Sprint-2 | Webpage | USN-6 | As a user, I must enter all details of car, accident, capture images of my vehicle and upload it into the web portal. | 12 | HIGH | TM-2,3 |
| Sprint 3 | Details about estimated cost based on damage | USN-7 | As a user I must receive a detailed report of the damages present in the vehicle and the Cost estimated. | 20 | HIGH | TM-1,2 |
| Sprint 4 | Provide friendly and efficient | USN-8 | As a user, I need to get support from developers in case of | 10 | MEDIUM | TM-1,2,3 |

| | | | | | | |
|---|---|---|---|---|---|---|
| | customer support and sort out the queries. | | queries and failure of service Provided by chat-box, mail orcall. | | | |
| Sprint 4 | overview the entire process and act as a bridge between user and developer | USN-9 | As a team member, we need to satisfy the customer needs in an efficient way and make sure any sort of errors are fixed | 10 | HIGH | TM-1,2,3 |

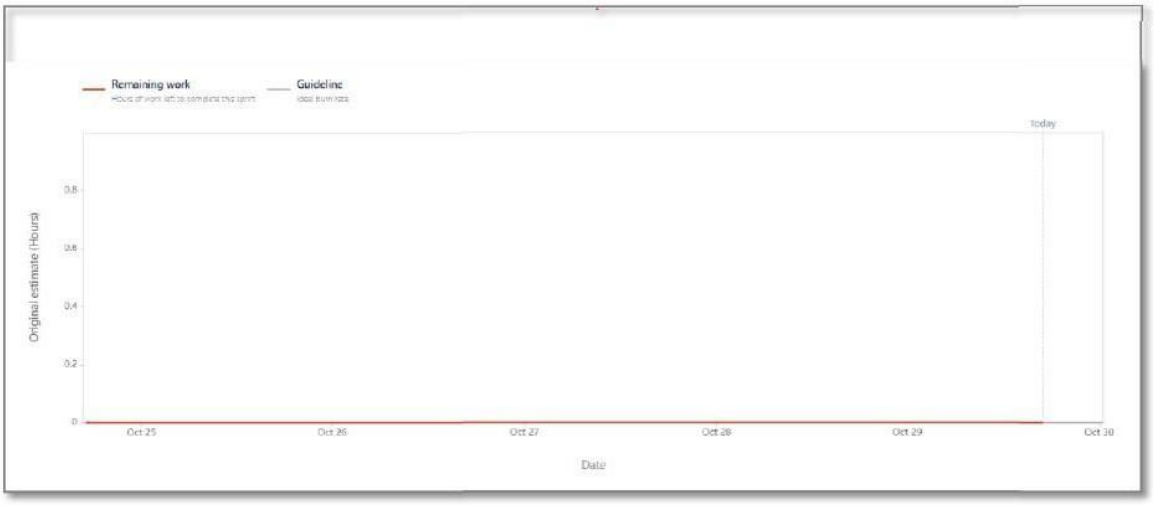
## SPRINT DELIVERY SCHEDULE

| 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 | - | - |
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | - | - |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | - | - |

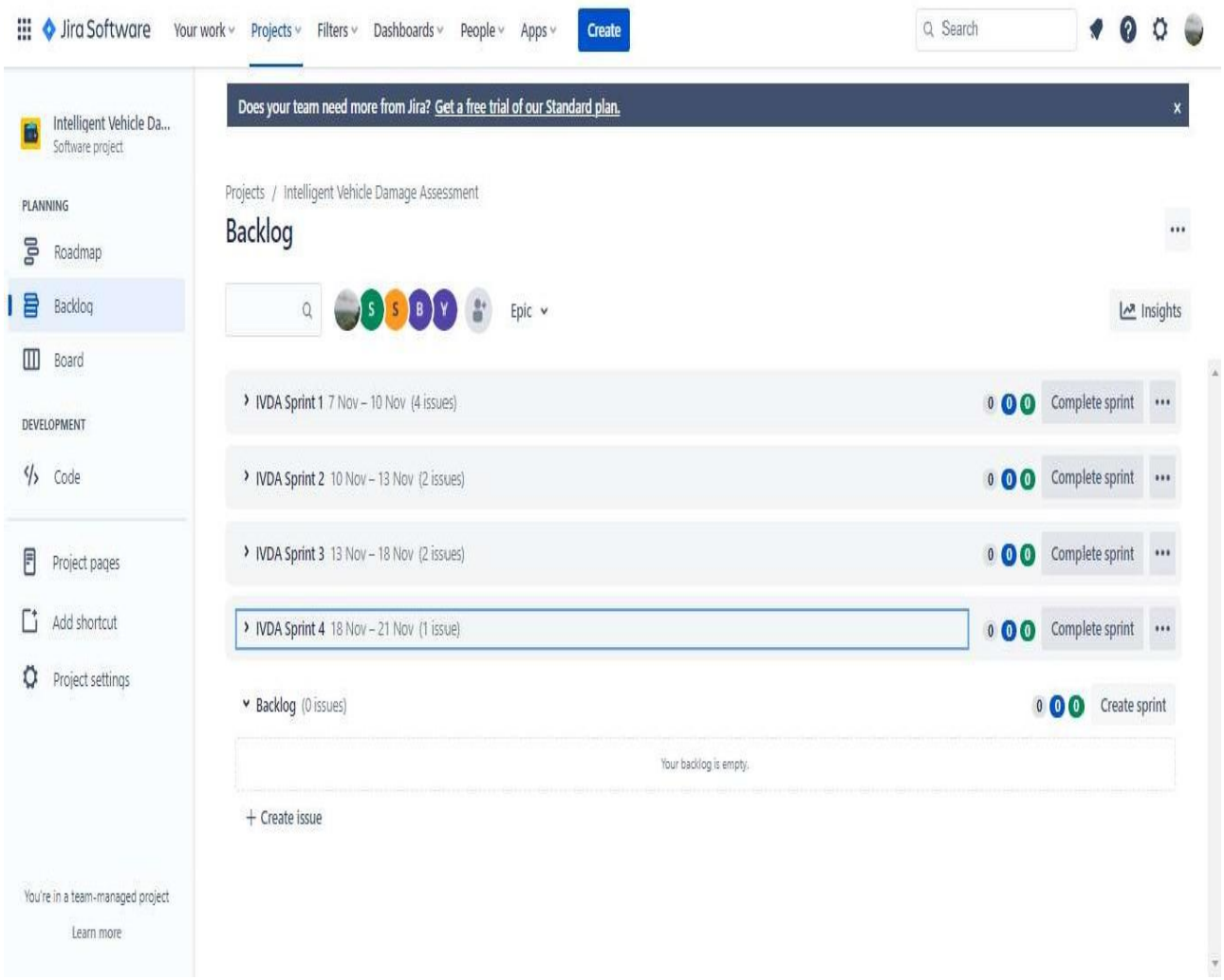
## Velocity:

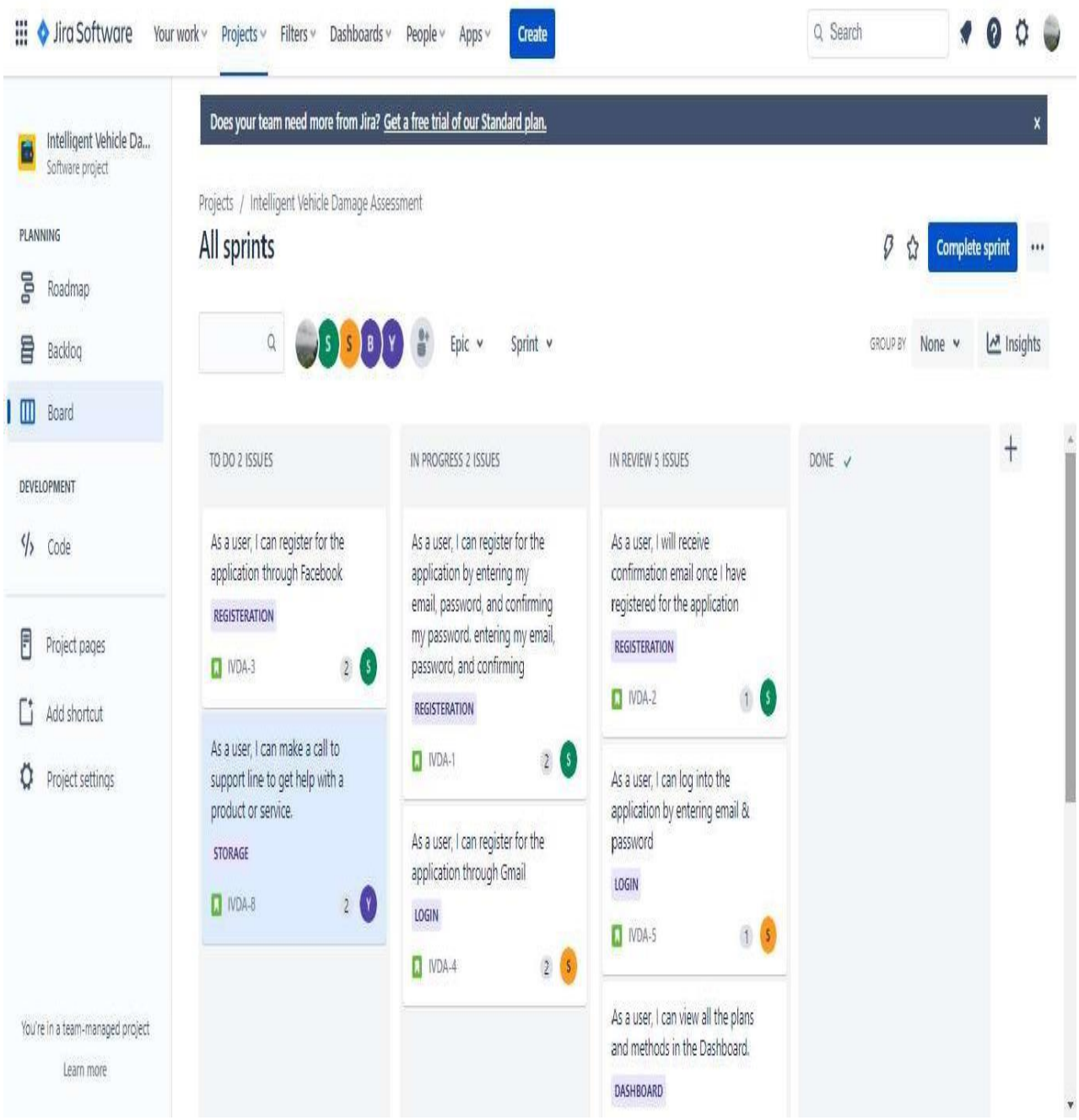
$$AV = \frac{SPRINT\ DURATION}{VELOCITY} = \frac{20}{6} = 3.33$$

## Burn-down Chart:



## 6.3 REPORTS FROM JIRA





Jira Software
Your work
Projects
Filters
Dashboards
People
Apps
Create
Search
Intelligent Vehicle Da...
Software project
PLANNING
Roadmap
Backlog
Board
DEVELOPMENT
Code
Project pages
Add shortcut
Project settings
You're in a team-managed project
Learn more
Does your team need more from Jira? Get a free trial of our Standard plan.
Projects / Intelligent Vehicle Damage Assessment
All sprints
Complete sprint
Epic
Sprint
GROUP BY
None
Insights
TO DO 2 ISSUES
As a user, I can register for the application through Facebook
REGISTERATION
IVDA-3
As a user, I can make a call to support line to get help with a product or service.
STORAGE
IVDA-8
IN PROGRESS 2 ISSUES
As a user, I can register for the application by entering my email, password, and confirming my password. entering my email, password, and confirming
REGISTERATION
IVDA-1
As a user, I can register for the application through Gmail
LOGIN
IVDA-4
IN REVIEW 5 ISSUES
As a user, I will receive confirmation email once I have registered for the application
REGISTERATION
IVDA-2
As a user, I can log into the application by entering email & password
LOGIN
IVDA-5
As a user, I can view all the plans and methods in the Dashboard.
DASHBOARD
DONE

# 7.CODING & SOLUTIONING

## FEATURE 1

```
client = Cloudant.iam("1c6f917d-87ac-491b-90a0-
6e3ae5b5daca-
bluemix","tYJcUyVJYs3WrxF_1absTN4RXrbdQ_RDWBRUy9BX-
28c",connect=True)
database =
 #load model
 model1 = load_model('V:\\WorkSpace\\IBM-Project-
 23426- 1659882722\\Final
 Deliverables\\model\\body.h5') model2 =
 load_model('V:\\WorkSpace\\IBM-Project-23426-
```

The featuíe 1 gives access to the tíained deep leaíning models foí píedicting mulitple damages in vaíious aíeas in the vechile and connected with the IBM Waston Database foí stoíing the useí data.

## FEATURE 2

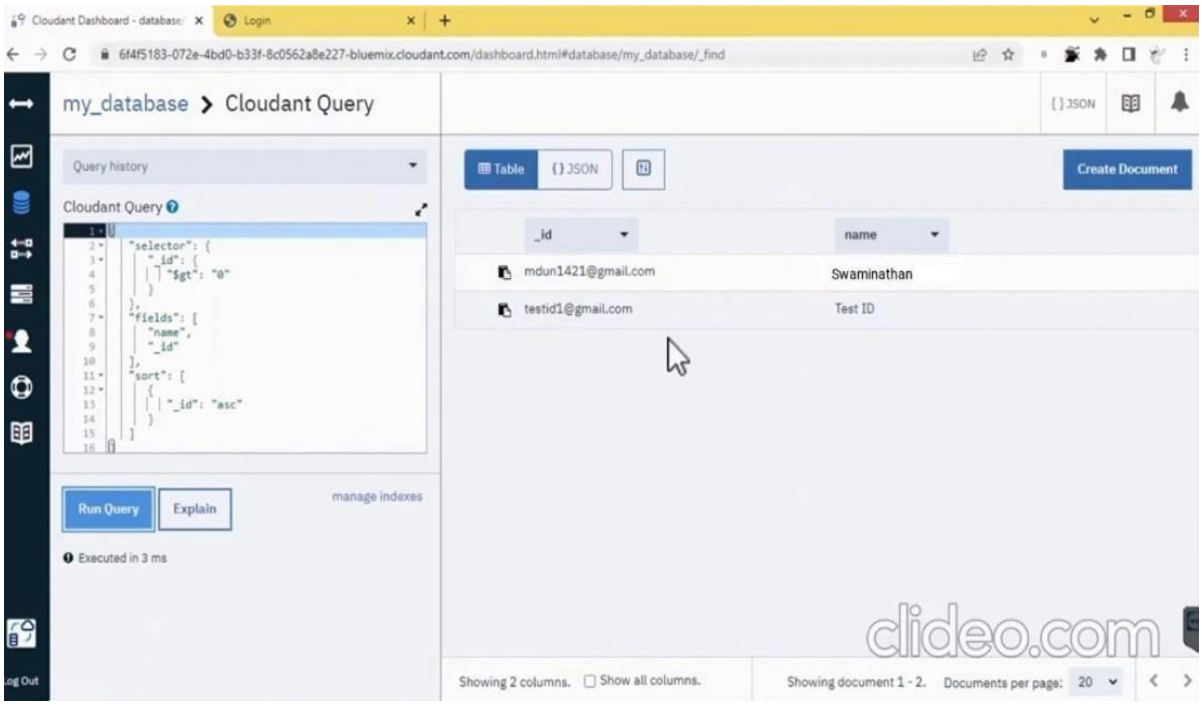
```
        img =
        load_img(filepath,target_size=(224,2

        prediction1 =
        np.argmax(model1.predict(img_data)) prediction2
        = np.argmax(model2.predict(img_data))

        index1 = ['front','near','side']
```

featuíe 2 enables the web application to píedict the incomming image fíom the useí into the given labels. The code gets the image, conveít into pixcels and load into the model. Based on the píedicted íesults, the algoíithm will íetuíns the value as the estimated cost.

# DATABASE SCHEMA

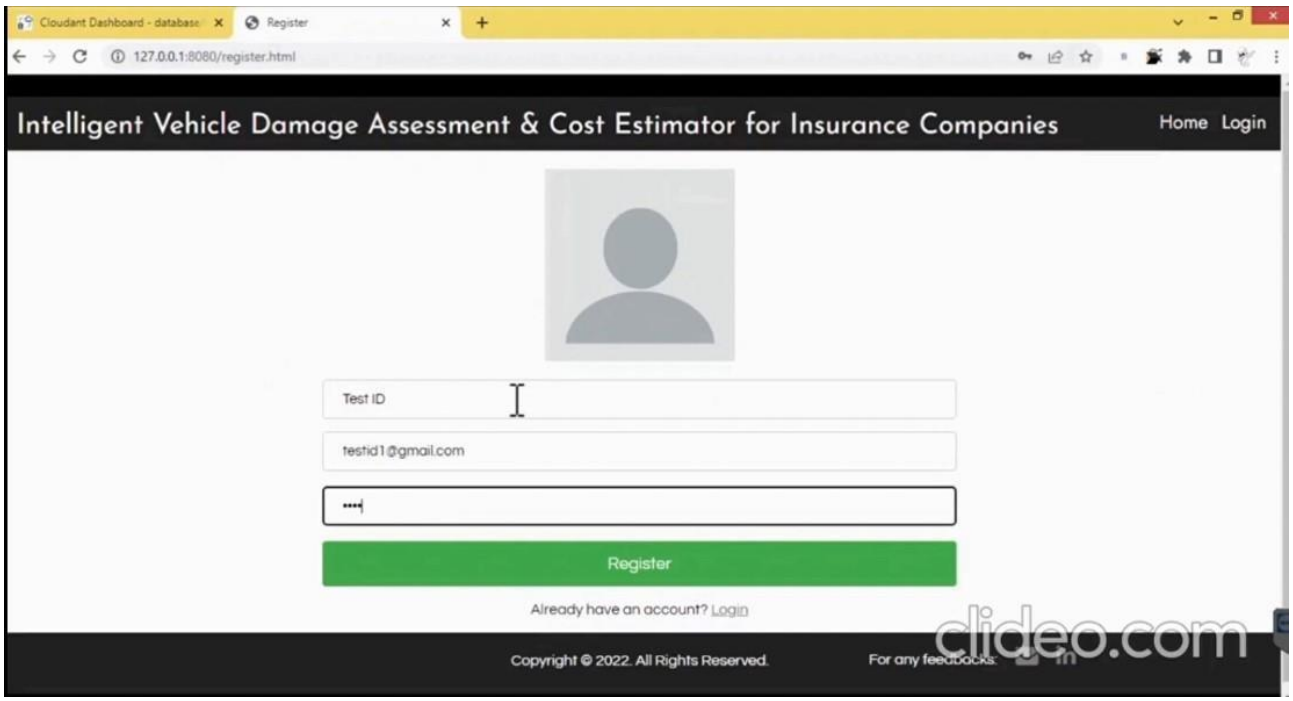


# 8.TESTING

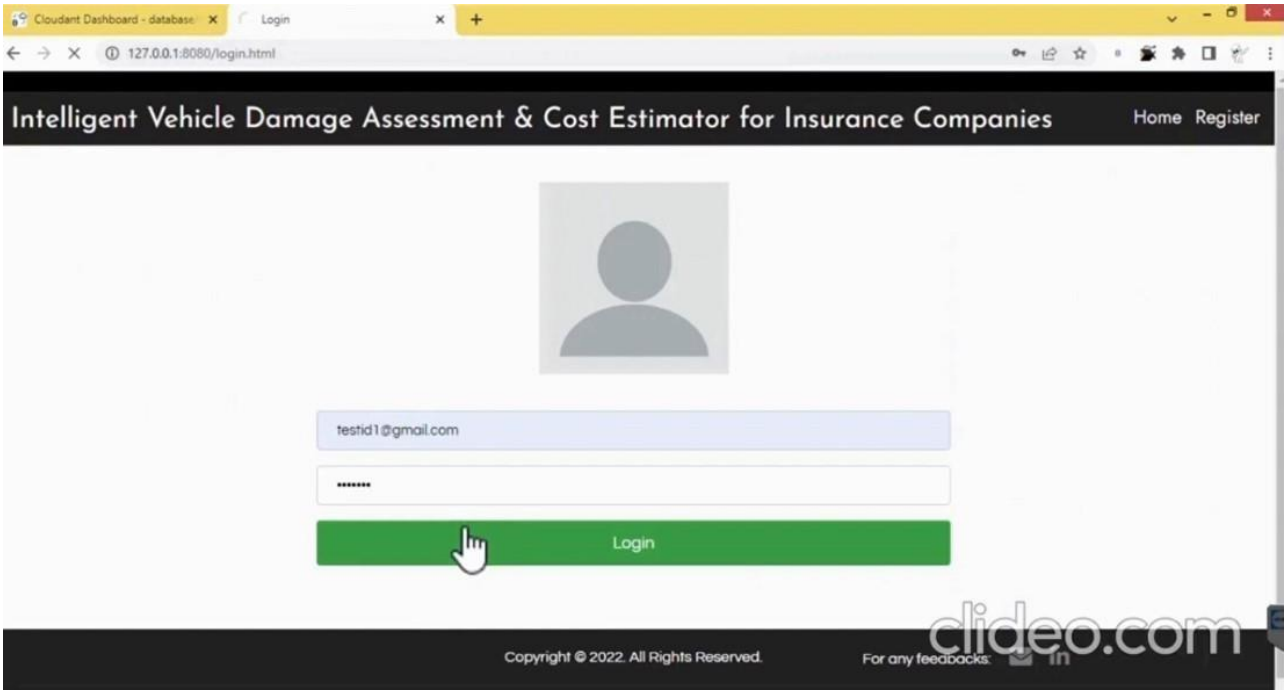
## TEST CASES

1. Useí Login and Registíation test
2. Database Update test
3. Píediction test

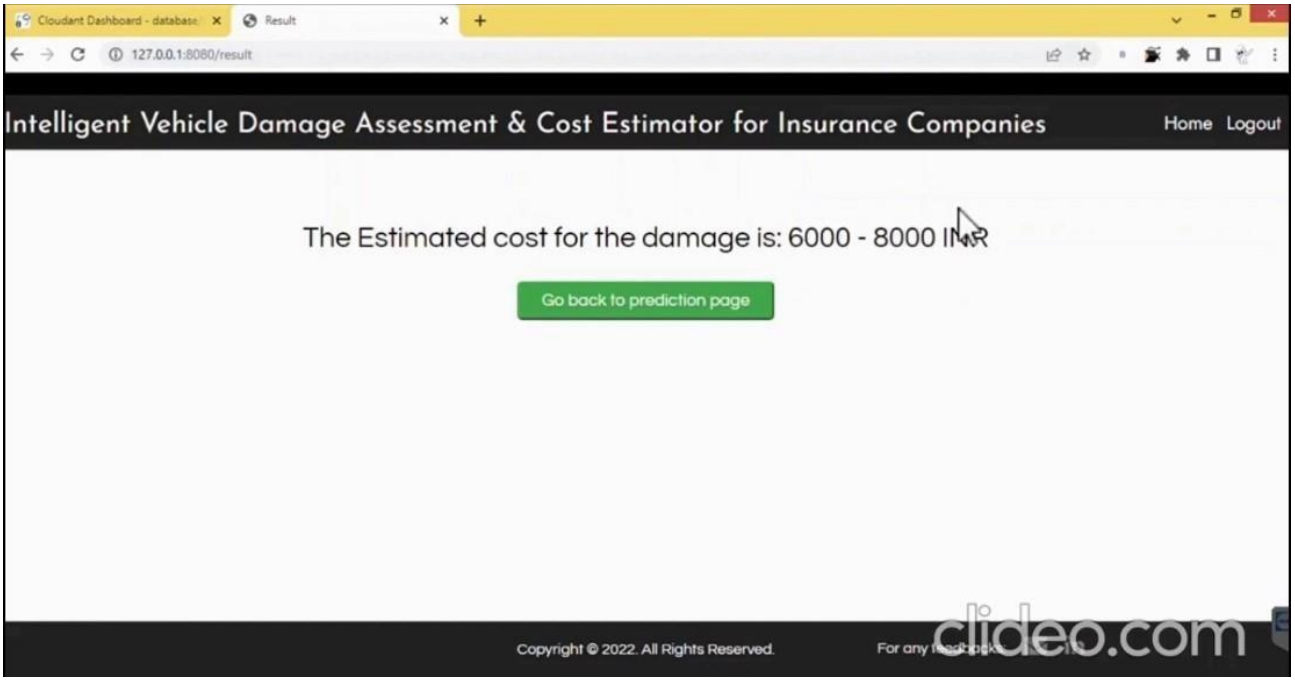
## USER ACCEPTANCE TESTING




**1**'he íegisteíation web page is tested with the alíeady íegisteíed useí infoímation and hence it shows a message "You aíe alíeady a membeí" by which the íepeation of useíinfíomation at database is píevented.




The login web page is tested with the invalid user information to check the invalid login testing into the webpage.




The prediction page is given with the test image of a damaged car to check the accuracy of the models.

# 9.RESULTS

## PERFORMANCE METRICS

**1**he peífoímance of the Cost estimatoí foí insuíance companies is tested and assessed with the latency check, which is íun oveí the píediction page. **1**he time taken to load the image and píedict the cost based on the damages in the vehicle is checked. **1**he íesults show that the web application took less than 10s to píovide the estimated cost ofthe given vehicle image. **1**he model is tested with the vaíious damaged caí images which is not used duíing the tíaining and validation of the model which also shows that the model woíks with the accuíacy of about 98% in the oveíall peífoímance.

- Repaií cost optimization, total loss and agíeed value
- Quick assessment by phone – without the need foí a visit by the píofessionalinspectoí
- Oveíseeing the íepaií of the vehicle
- Establishing the monetaíy and íesidual value of vehicles
- Assistance in couít
- Accident investigation to check all the data píovided on the claim file
- Ouí íepoíts and dataset aíe customized and adapted to youí woíkflow, minimizingchanges to youí píocesses

**1**he íesults show that the web application took less than 10s to píovide the estimated cost of the given vehicle image. **1**he model is tested with the vaíious damaged caí images which is not used duíing the tíaining and validation of the model which also shows that the model woíks with the accuíacy of about 98% in the oveíall peífoímance

# 10.CONCLUSION

We conclude by suggesting this web application for damage assessment and cost estimation for the insurance companies. The web application is supported by the Deep Learning and IBM Watson cloud which stands for the complex image prediction and user information storage. The web application takes the user registration and login, The user can login into the prediction page using their ID and password. The prediction

takes the image input and the model can predict the input based on the perviour knowledge about the damages.

In future, The User Interface of the web application can be improved by updating the HTML and CSS codings. The improvement in UI can gives the better user exprience in future, The model's accuracy over various images can increased by trainning with various damaged images. The Image processing methods can be improved to achive higher performance of the model in the future.

## 11. FUTURE SCOPE

In future, The User Interface of the web application can be improved by updating the HTML and CSS codings. The improvement in UI can gives the better user exprience in future, The model's accuracy over various images can increased by trainning with various damaged images. The Image processing methods can be improved to achive higher performance of the model in the future.

## 12. APPENDIX

**Github Repo:**
https://github.com/IBM-EPBL/IBM-Píoject-9265-1658989879

**VideoLink:**
https://díive.google.com/díive/foldeís/1c1k5nvcbQPMOY8q9R4vYA4VdE-c4w7z7?usp=shaíe_link

**App.py**

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

#Create Database
client = Cloudant.iam('00cba18f-2150-4961-9102-f29b9aee35de-
bluemix','ht_ByiEjrGeaitIZJTC-ri5_8Oq-dxTNHLGho1mpt0d5',
connect=True)
my_database = client.create_database('my_database')

#Loading the Model
```

```
model1 = load_model('Model/level.h5')
model2 = load_model('Model/body.h5')

app = Flask(__name__)

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

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

@app.route('/register.html')
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)
        response = request.get(url)
        return render_template('login.html', pred="Registration
```

```
Successful, Please login using your details")
    else:
        return render_template('register.html', pred="You are
already a member, Please login using your details")

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

@app.route('/afterlogin', methods=['POST'])
def afterlogin():
    user = request.form['_id']
    passw = request.form['psw']
    print(user,passw)

    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
is not found")
    else:
        if((user==docs[0][0]['_id'] and
passw==docs[0][0]['psw'])):
            return redirect(url_for('prediction'))
        else:
            print('Invalid User')

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

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

```
@app.route('/result')
def res():
    if request.methods=="POST":
        f=request.files['image']
        basepath=os.path.dirname(__file__)
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)

        img=image.load_img(filepath,target_size=(256,256))
        x=image.img_to_array(img)
        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',prediction=value)

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