

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

Purpose

## **2 LITERATURE SURVEY**

Existing problem

References

Advantages & Disadvantages

## **3 IDEATION & PROPOSED SOLUTION**

Empathy Map Canvas

Ideation & Brainstorming

Proposed Solution

Problem Solution fit

## **4 REQUIREMENT ANALYSIS**

Functional requirement

Non-Functional requirements

## **5 PROJECT DESIGN**

Data Flow Diagrams

Solution & Technical Architecture

User Stories

## **6 PROJECT PLANNING & SCHEDULING**

Sprint Planning & Estimation

Sprint Delivery Schedule

Reports from JIRA

## **7 CODING & SOLUTIONING**

Feature 1

Feature 2

Database Schema (if Applicable)

## **8 TESTING**

Test Cases

User Acceptance Testing

## **9 RESULTS**

Performance Metrics

## **10 CONCLUSION**

## **11 FUTURE SCOPE**

## **12 APPENDIX**

# 1. INTRODUCTION

## **Project Overview**

The project "Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies" is a responsive web application powered by artificial 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 database 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 Watson 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 flask which is a python framework for designing 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 database of the IBM Cloud, during the time of login, the login ID is compared with the ID in the database and allow the user to the next page. The Deep Learning model is built using the VGG16 which is present in the Keras library and the model is trained with the images of multiple cars with various levels and types of damages. The model is deployed in the back-end using the flask and the prediction.html page is set to collect the image from the user. The prediction algorithm is used to treat the image and estimate 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 built using HTML and CSS which is connected to back - end which is built 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 Watson Cloud, a responsive web application may be created. The database API key is gathered and connected with the front-end using flask, which is a python framework for designing the backend.

## 2.LITERATURE SURVEY

TITLE	AUTHOR	YE AR	DESCRI PTION	ADVAN TAGES	DISADVA NTAGES	METHOD OLOGY
The use of telematics data in vehicle insurance	<ul style="list-style-type: none"> <li>K.Korishchenko</li> <li>I. Stankevich</li> <li>N. Pilnik</li> <li>D. Petrova</li> </ul>	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, Increased 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 construction	<ul style="list-style-type: none"> <li>Jihabqaddour</li> <li>Syeda Ayesha siddiq</li> </ul>	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 synthetically 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	<ul style="list-style-type: none"> <li>Qianqian Zhu<sup>1</sup>,</li> <li>Wei Hu<sup>2</sup>,</li> <li>Yingnan Liu<sup>1</sup> and Zihao Zhao<sup>1</sup></li> </ul>	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	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.
Damage d Vechile parts Recognit ion Using Capsule Neural Network	KundjanasithThonglek NorawitUrailertprasertPatcharaPattiyathan eeChantanaChantrapornchai	2022	This paper shows about Damgedv echile part recgonitio n using capsule neural network.a damaged vehicle part detection platform, called Intelligent Vehicle Accident Analysis (IVAA) which provides artificial intelligenc e as a service (AlaaS), is proposed.	damage localizatio n and damage classificati on. The accuracy of the damage localizatio n is 93.28% and the accuracy of the damage classificati on is 98.47%, respective ly.	error-prone, time-consuming, and requires man-hour workers.	The system helps automaticall y 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	Adrita Barari N.V.S Abhilash Payanshi Jain Ankit Sati Karthik Sai Datta 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 GreetaKavithaJayaraj MehaaKanakarajan D. Malathi	2021	In this paper, they propose employing convolutional 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 GuoWeiming Shen Ying 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.  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.

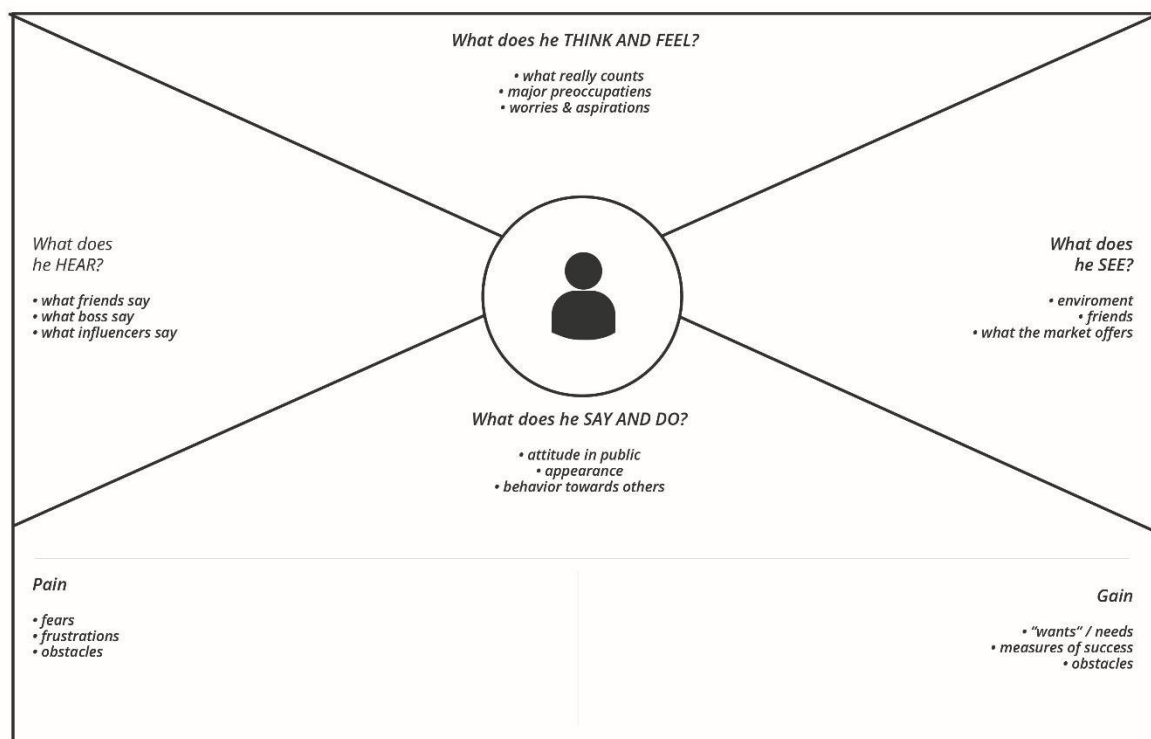
S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	In this project , we use car images for detecting the damages and calculated the associated cost
2.	Idea / Solution description	Computer Vision and Deep Learning techniques to accurately classify vehicle damage to facilitate claims 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 help 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:

#### Empathy Map



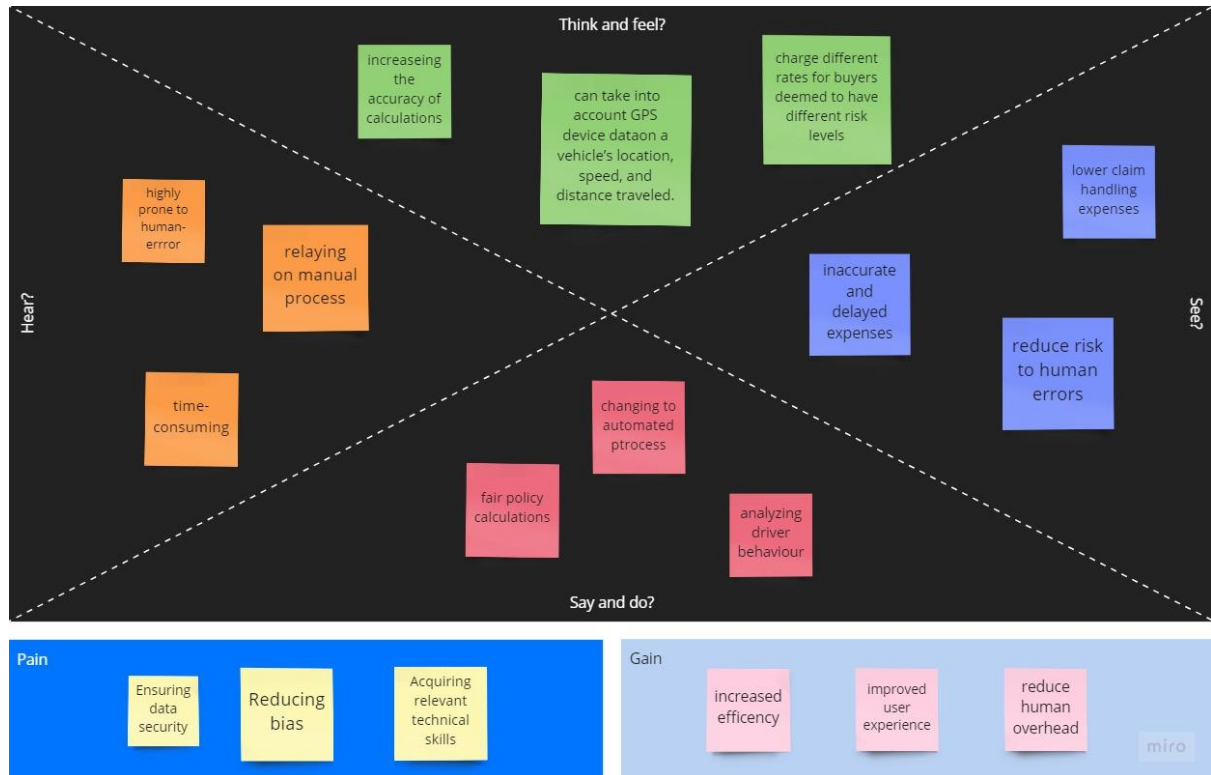
<http://creativecommons.org/licenses/by-sa/4.0/>

Business Model **Toolbox**

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



## Brainstorm & idea prioritization

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.

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



### 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 [your problem statement]?



### Key rules of brainstorming

To run a smooth and productive session

- 🗣️ Stay in topic. 💡 Encourage wild ideas.
- ⏸️ Defer judgment. 👂 Listen to others.
- 🗣️ Go for volume. 👁️ If possible, be visual.

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

P.Sree Likithaa

the use of telematics data in vehicle insurance.

we can use well-organized deep knowledge-based constructions for detecting, localizing, and classification vehicle damage using enhanced Mask R-CNN method which integrates deep learning, instance segmentation.

Sowmiya.R

we can use deep learning target detection method to find the damaged parts in vehicle.

we use different damage determination algo for determining the type and depth of the knowledge.

Sowmiyasree

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

The damages can be assessed by granular part detection, damage localization and classification into different damage types

Revathi

vehicle appearance damage recognition algorithm based on deep learning and its model evaluation method .

Damaged vechile parts recognition ising capsule neural network (damage localization and damage classication)

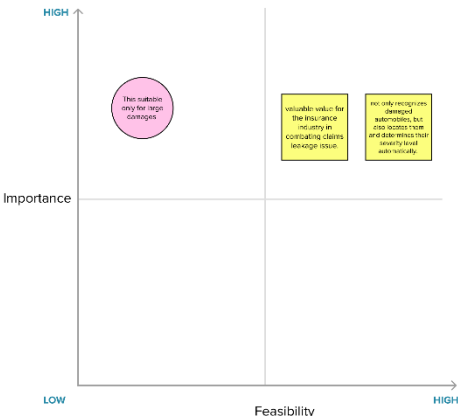
# Step-3: Idea Prioritization

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



## 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.



# Proposed Solution It

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

Project Design Phase-I - Problem Solution Fit

Team ID: PNT2022TMID00513

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> <small>Who is your customer? i.e. working parents of 0-5 y.o. kids</small>  <b>Drivers aged between 25 and 65</b> are the most common age group of customers for car insurance.	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> <small>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</small>  <b>Customers may feel that our website is <u>not trustworthy</u> due to some other scam websites.</b>	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> <small>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 &amp; cons do these solutions have? i.e. pen and paper is an alternative to digital notetaking</small>  <b>Just by sending the image of damaged car to our website, customer gets the details of amount to be <u>claimed in a minute</u> rather than days if it is inspected visually. There won't be any claim leakage problems.</b>	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> <small>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.</small>  <b>Customers limit themselves from claiming insurance for minor damages because of <u>claims leakage</u> (Difference between the final settled amount paid out by an insurer and the amount that they could've paid had the claims process been more efficient).</b>	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span> <small>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</small>  <b>The real problem arises when the customer has severe damage on the car and they get <u>minimum amount</u> than expected. Since many people are involved at various stages of a claim, there is lack of visibility which makes the process to <u>slow down</u> and <u>over-complicated</u> at different stages.</b>	<b>7. BEHAVIOUR</b> <span>BE</span> <small>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)</small>  <b>Whenever the customer has damage on the car, they <u>meet the insurer</u> and apply for claim amount. As this process is time consuming, the customers <u>search</u> for car insurance websites to claim the amount. They upload the image of damaged car and get the details of claim amount within fraction of seconds.</b>	
Focus on J&P, tap into BE, understand RC	<b>3. TRIGGERS</b> <span>TR</span> <small>What triggers customers to act? i.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.</small>  <b>Being <u>transparent</u> to the customers by not making any false guarantees</b>	<b>10. YOUR SOLUTION</b> <span>SL</span> <small>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.</small>  <b>The aim of this project is to estimate the cost of damaged car accurately by detecting the area of damage, categorizing the damage with precision in a <u>fast and intelligent manner</u>. It can be used by insurance companies for faster processing of claims if users can upload pictures.</b>	<b>8. CHANNELS of BEHAVIOUR</b> <span>CH</span> <b>8.1 ONLINE</b> <small>What kind of actions do customers take online? Extract online channels from #7</small> <ul style="list-style-type: none"> <li>➤ Select the model of damaged car.</li> <li>➤ Select the city where you live.</li> <li>➤ Upload the image of damaged</li> </ul> <b>8.2 OFFLINE</b> <small>What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development.</small> <ul style="list-style-type: none"> <li>➤ Meeting the insurer.</li> <li>➤ Filling application forms.</li> <li>➤ Submitting the required documents.</li> </ul>	Focus on J&P, tap into BE, understand RC
	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> <small>How do customers feel when they face a problem or a job and afterwards? i.e. lost, insecure &gt; confident, in control - use it in your communication strategy &amp; design.</small>  <b>We should prove that our website is better than others by providing good customer support, gaining the customer trust and provide customer satisfaction.</b>			
Identify strong TR & EM				Identify strong TR & EM

## 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 Registration through Gmail
FR-2	User Confirmation	Confirmation via Email 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.

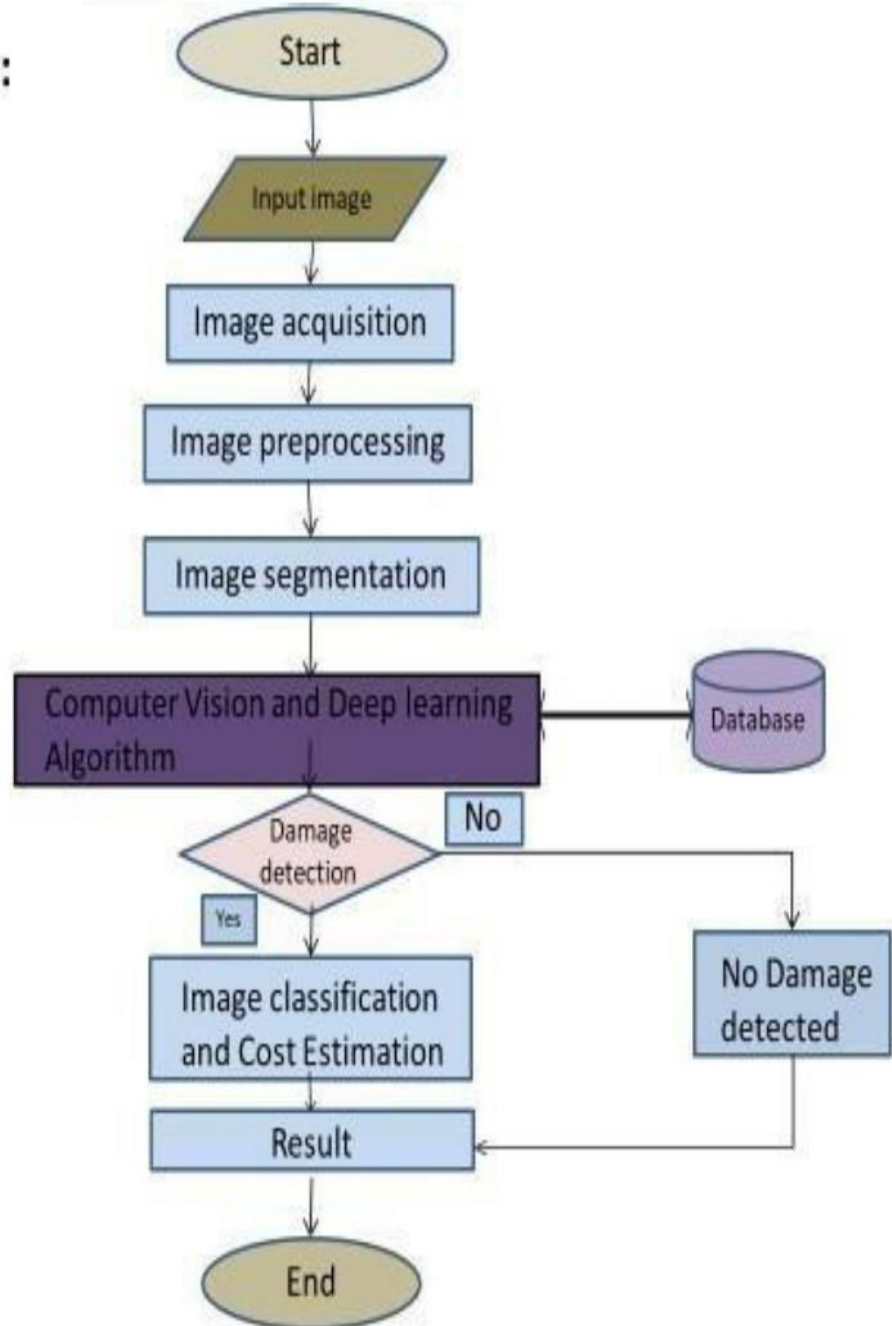
FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	More efficient for the frequent users. users can easily understand what the application does and feel satisfied with the system.
NFR-2	<b>Security</b>	<ul style="list-style-type: none"><li>• AI powered vehicle damage assessment and cost estimator for insurance company should contain more security in which our data which entered or maintained should be more security.</li><li>• With the help of the username and password it provides more security in which it can access more securable and the data are private</li></ul>
NFR-3	<b>Reliability</b>	This application must perform without failure in 90 percentage of use cases during a month. It is morereliable.
NFR-4	<b>Performance</b>	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	<b>Availability</b>	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	<b>Scalability</b>	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 improved to:

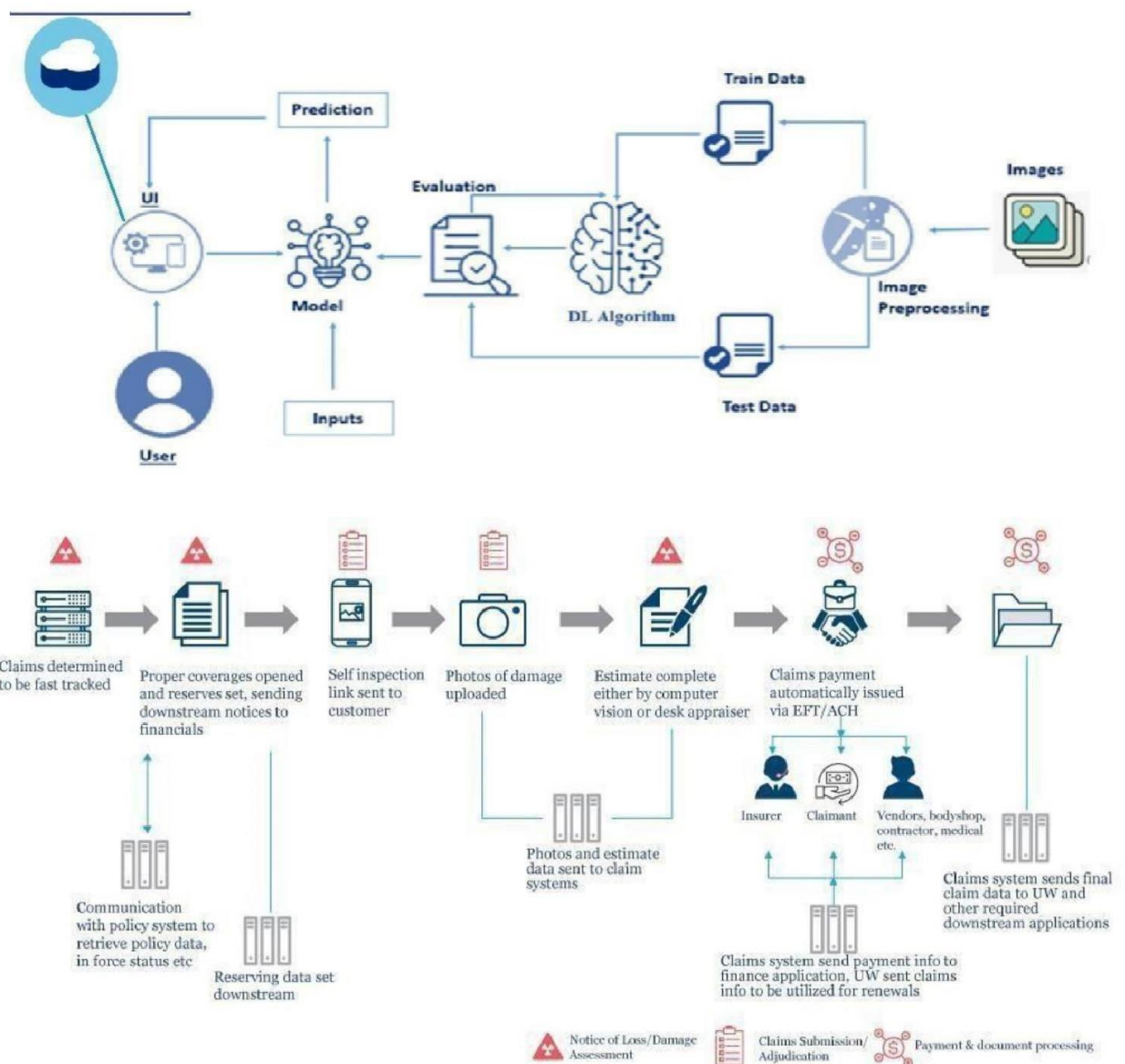
1. Obtain a cost estimate
2. Send assessment to insurance carrier
3. Print documentation

## **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:

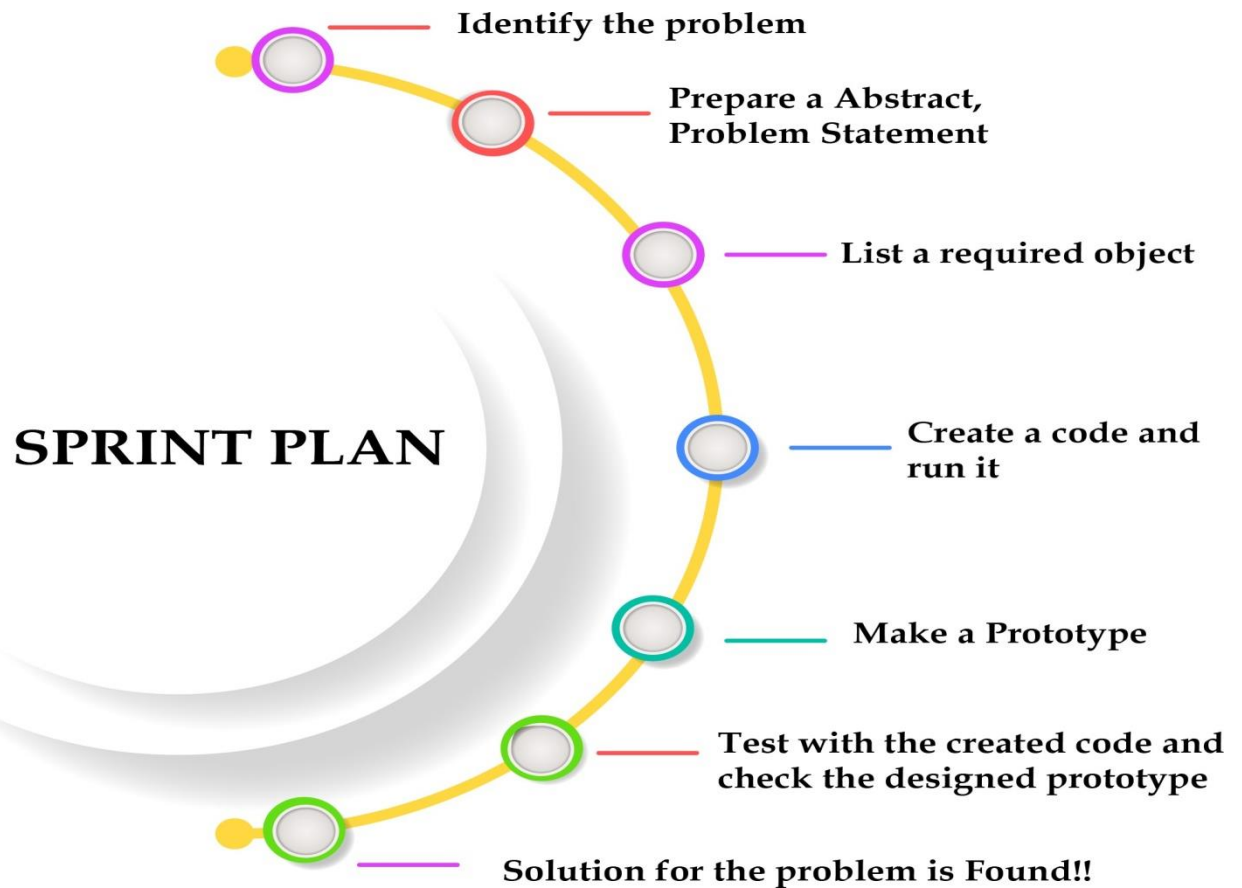


## USER STORIES

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

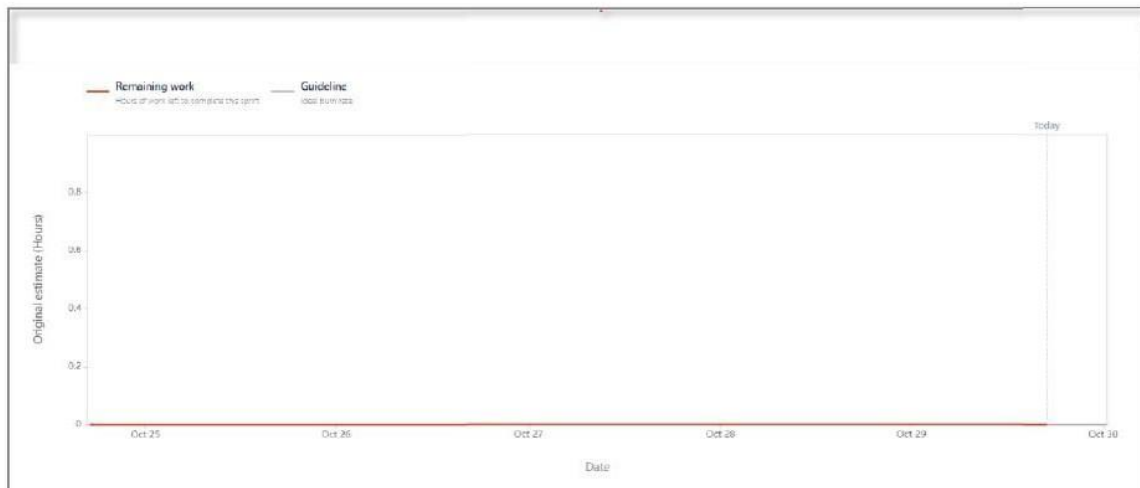
## SPRINT DELIVERY SCHEDULE

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

Your work ▾ Projects ▾ Filters ▾ Dashboards ▾ People ▾ Apps ▾ 

Create

Q Search

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

Backlog

Q

Epic ▾

IVDA Sprint 1 7 Nov – 10 Nov (4 issues)000Complete sprint...

IVDA Sprint 2 10 Nov – 13 Nov (2 issues)000Complete sprint...

IVDA Sprint 3 13 Nov – 18 Nov (2 issues)000Complete sprint...

IVDA Sprint 4 18 Nov – 21 Nov (1 issue)000Complete sprint...

▼ Backlog (0 issues)000Create sprint

Your backlog is empty.

+ Create issue

	T	NOV	DEC	JAN '23
Sprints		<div>IVDA-10 IVDA-11</div>		
>  IVDA-10 Registration		<div></div>		
>  IVDA-11 Login		<div></div>		
>  IVDA-12 Dashboard		<div></div>		
>  IVDA-13 Storage		<div></div>		
>  IVDA-14 Output		<div></div>		



## 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



Q



Epic

Sprint

GROUP BY

None

Insights

## TO DO 2 ISSUES

As a user, I can register for the application through Facebook

REGISTRATION

IVDA-3

2



As a user, I can make a call to support line to get help with a product or service.

STORAGE

IVDA-8

2



## 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

REGISTRATION

IVDA-1

2



As a user, I can register for the application through Gmail

LOGIN

IVDA-4

2



## IN REVIEW 3 ISSUES

As a user, I will receive confirmation email once I have registered for the application

REGISTRATION

IVDA-2

1



As a user, I can log into the application by entering email &amp; password

LOGIN

IVDA-5

1



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
    modell = load_model('V:\\Workspace\\IBM-Project-23426- 1659882722\\Final
6 Deliverables\\model\\body.h5') model2 =
    load_model('V:\\Workspace\\IBM-Project-23426-
```

The feature 1 gives access to the trained deep learning models for predicting multiple damages in various areas in the vehicle and connected with the IBM Watson Database for storing the user data.

### FEATURE 2

```
1         img =
2         load_img(filepath,target_size=(224,2
3
4         prediction1 =
5         np.argmax(model1.predict(img_data)) prediction2
6         = np.argmax(model2.predict(img_data))
7
8         index1 = ['front','near','side']
```

feature 2 enables the web application to predict the incoming image from the user into the given labels. The code gets the image, convert into pixels and load into the model. Based on the predicted results, the algorithm will returns the value as the estimated cost.

# DATABASE SCHEMA

The screenshot displays the Cloudant Dashboard interface for a database named 'my\_database'. The left sidebar contains navigation icons for various database functions. The main area is titled 'my\_database > Cloudant Query'. It features a 'Query history' dropdown, a 'Cloudant Query' section with a query editor, and buttons for 'Run Query', 'Explain', and 'manage indexes'. The query editor contains a JSON query: 

```
{
  "selector": {
    "id": {
      "$gt": "a"
    }
  },
  "fields": [
    "name",
    "_id"
  ],
  "sort": [
    { "_id": "asc" }
  ]
}
```

. Below the query editor, it indicates 'Executed in 3 ms'. The right side of the dashboard shows the query results in a table view. The table has two columns: '\_id' and 'name'. The results are as follows:

_id	name
mdun1421@gmail.com	Swaminathan
testid1@gmail.com	Test ID

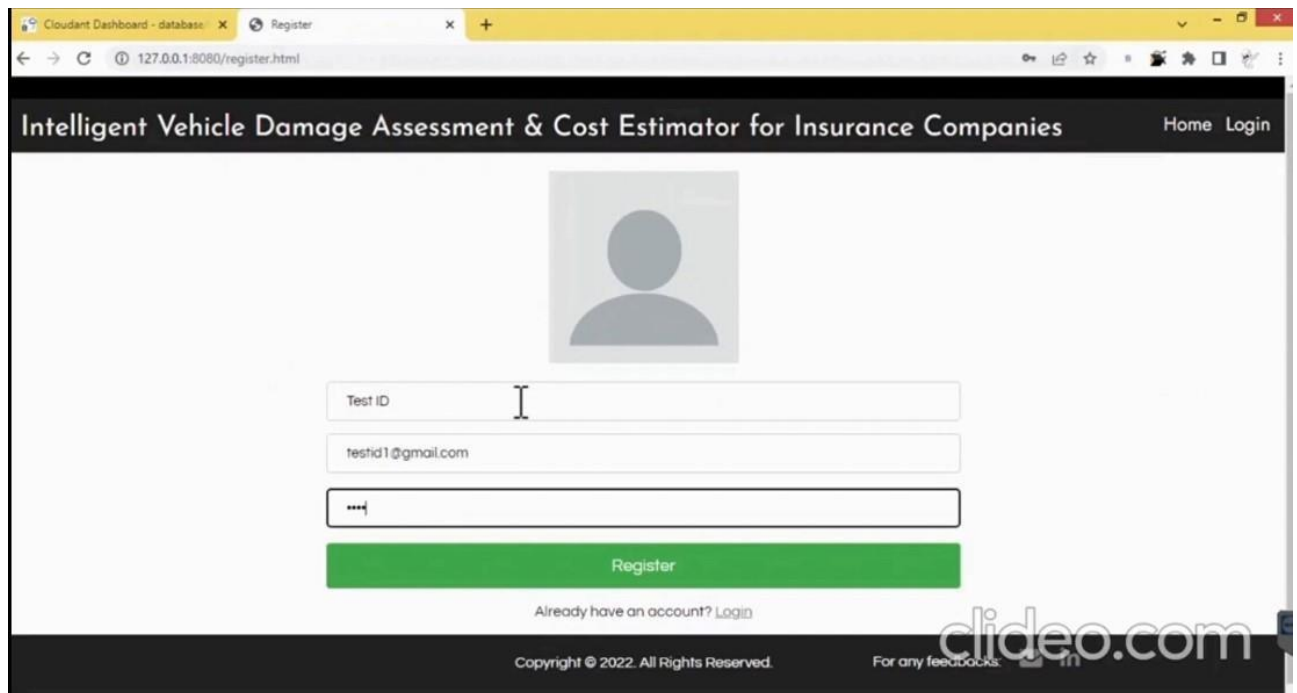
At the bottom of the dashboard, there is a status bar showing 'Showing 2 columns.', a checkbox for 'Show all columns.', and pagination information: 'Showing document 1 - 2. Documents per page: 20'.

## 8.TESTING

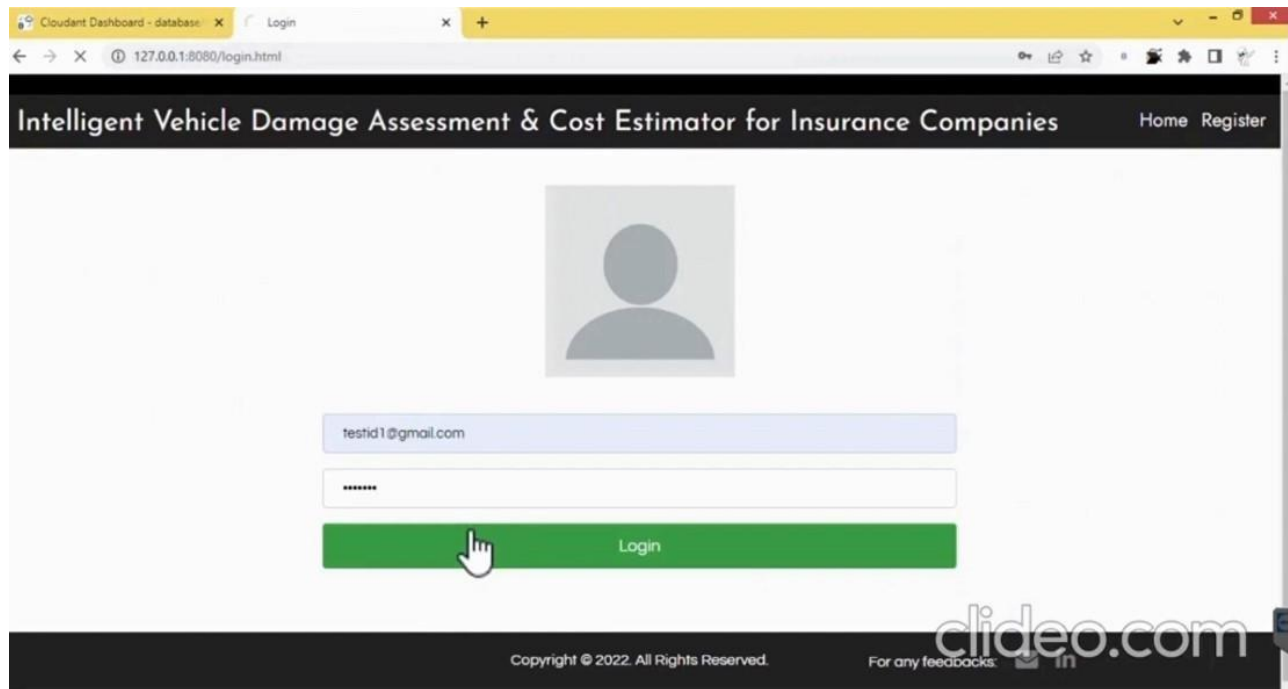
### TEST CASES

1. User Login and Registration test
2. Database Update test
3. Prediction test

### USER ACCEPTANCE TESTING

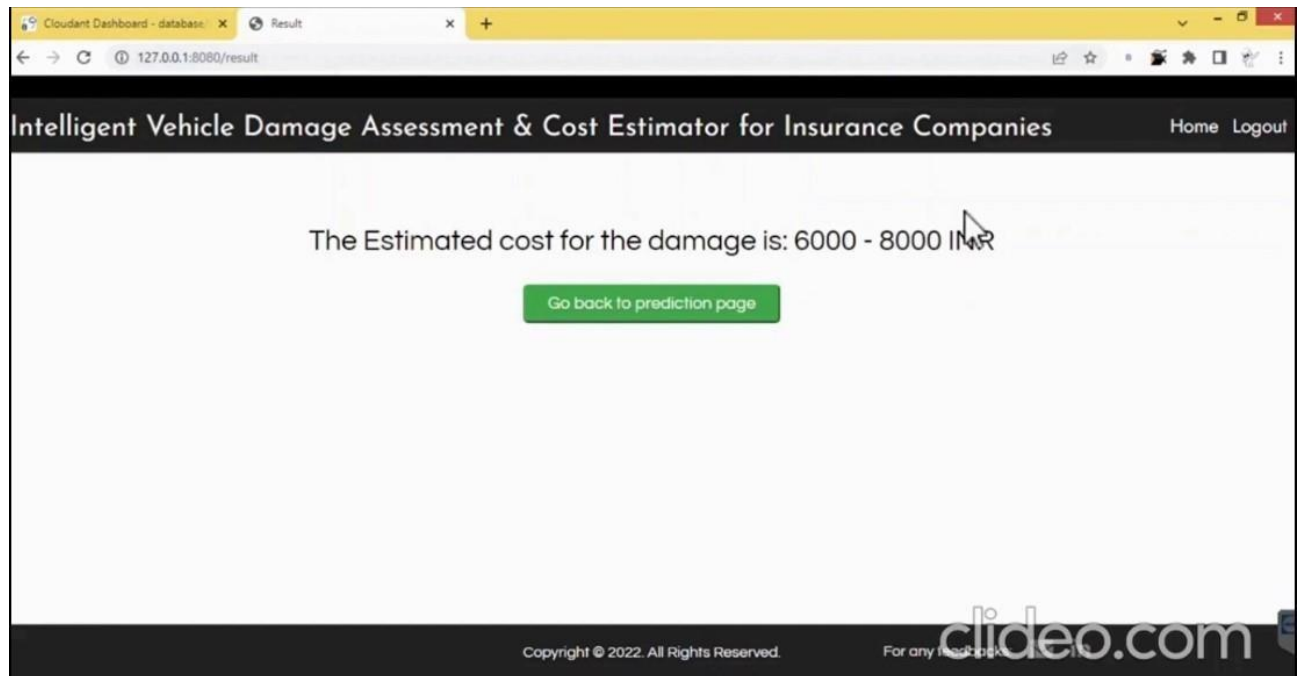


The registration web page is tested with the already registered user information and hence it shows a message "You are already a member" by which the repetition of user information at database is prevented.



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

The performance of the Cost estimator for insurance companies is tested and assessed with the latency check, which is run over the prediction page. The time taken to load the image and predict the cost based on the damages in the vehicle is checked. The results show that the web application took less than 10s to provide the estimated cost of the given vehicle image. The model is tested with the various damaged car images which is not used during the training and validation of the model which also shows that the model works with the accuracy of about 98% in the overall performance.

- Repair cost optimization, total loss and agreed value
- Quick assessment by phone – without the need for a visit by the professional inspector
- Overseeing the repair of the vehicle
- Establishing the monetary and residual value of vehicles
- Assistance in court
- Accident investigation to check all the data provided on the claim file
- Our reports and dataset are customized and adapted to your workflow, minimizing changes to your processes

The results show that the web application took less than 10s to provide the estimated cost of the given vehicle image. The model is tested with the various damaged car images which is not used during the training and validation of the model which also shows that the model works with the accuracy of about 98% in the overall performance

## **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 training 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 training 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-Project-9265-1658989879>

### VideoLink:

[https://drive.google.com/drive/folders/1c1k5nvcbQPMOY8q9R4vYA4VdE-c4w7z7?usp=share\\_link](https://drive.google.com/drive/folders/1c1k5nvcbQPMOY8q9R4vYA4VdE-c4w7z7?usp=share_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-dxTNHLGholmpt0d5',
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)
```



