

# **INTELLIGENT VEHICLE DAMAGE ASSESSMENT AND COST ESTIMATOR FOR INSURANCE COMPANY**

## **A PROJECT REPORT**

*Submitted by*

<b>NEETHIARASAN S</b>	<b>711719104056</b>
<b>NITHESH V</b>	<b>711719104060</b>
<b>SANJAY K</b>	<b>711719104079</b>
<b>VEERAKUMARAVELU M</b>	<b>711719104106</b>

*in partial fulfilment for the award of the degree*

*of*

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**KGISL INSTITUTE OF TECHNOLOGY, SARAVANAMPATTI**

**ANNA UNIVERSITY:: CHENNAI 600 025**

**APRIL 2022**

## **ABSTRACT**

Analysis of the damaged vehicle that can be automatically claiming insurance that takes human resource, time and effort. Image processing and machine learning techniques are analysing the vehicle damage in the proposed solution. The Advanced solution helps to speed up the claiming process sufficiently. Consider a situation, if a person is driving a car they met an accident the vehicle owner can taken a few photos of the damaged car from a mobile phone that can be send to the insurance company and can just upload the photos to the system. The system can analyse the damage, severity of the damage as well as location of the damage. In this proposed project the insurance company can machine-driven the car damage analysis process without the need for humans to analyse the damage done to the car. Therefore, it is a very challenging task for quality of computer vision techniques and also Machine learning technologies.

## **TABLE OF CONTENTS**

<b>CHAPTER NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	<b>ABSTRACT</b>	<b>ii</b>
	<b>LIST OF TABLES</b>	
	<b>LIST OF FIGURES</b>	
	<b>LIST OF ABBREVIATIONS</b>	<b>2</b>
<b>1.</b>	<b>INTRODUCTION</b>	
	1.1 PROJECT OVERVIEW	<b>3</b>
	1.2 PURPOSE	<b>4</b>
<b>2.</b>	<b>LITERATURE SURVEY</b>	
	2.1 EXISTING PROBLEM	<b>5</b>
	2.2 REFERENCES	<b>5</b>
	2.3 PROBLEM STATEMENT DEFINITION	<b>7</b>
<b>3.</b>	<b>IDEATION AND PROPOSED SOLUTION</b>	
	3.1 EMPATHY MAP CANVAS	<b>9</b>
	3.2 IDEATION AND BRAINSTORMING	<b>10</b>
	3.3 PROPOSED SOLUTION	<b>16</b>
	3.4 PROBLEM SOLUTION FIT	<b>19</b>
<b>4.</b>	<b>REQUIREMENT ANALYSIS</b>	
	4.1 FUNCTIONAL REQUIREMENTS	<b>20</b>
	4.2 NON-FUNCTIONAL REQUIREMENTS	<b>21</b>
<b>5.</b>	<b>PROJECT DESIGN</b>	
	5.1 DATA FLOW DIAGRAM	<b>23</b>
	5.2 SOLUTION & TECHNICAL ARCHITECTURE	<b>24</b>
	5.3 USER STORIES	<b>25</b>
<b>6.</b>	<b>PROJECT PLANNING AND SCHEDULING</b>	
	6.1 SPRINT PLANNING AND ESTIMATION	<b>29</b>
	6.2 SPRINT DELIVERY SCHEDULE	<b>31</b>
<b>7.</b>	<b>CODING AND SOLUTIONING</b>	
	7.1 FEATURE 1	<b>32</b>
	7.2 FEATURE 2	<b>32</b>

<b>8.</b>	<b>TESTING</b>	<b>33</b>
	8.1 TEST CASES	
	8.2 USER ACCEPTANCE TESTING	
	8.2.1 Defect Analysis	<b>35</b>
	8.2.2 Test Case Analysis	<b>35</b>
<b>9.</b>	<b>RESULTS</b>	
	9.1 PERFORMANCE METRICS	<b>36</b>
<b>10.</b>	<b>ADVANTAGES AND DISADVANTAGES</b>	<b>37</b>
<b>11.</b>	<b>CONCLUSION</b>	<b>39</b>
<b>12.</b>	<b>FUTURE SCOPE</b>	<b>40</b>
<b>13.</b>	<b>APPENDIX</b>	
	13.1 SOURCE CODE	<b>41</b>
	13.2 GITHUB AND PROJECT DEMO	<b>62</b>
<b>14.</b>	<b>REFERENCES</b>	<b>63</b>

## **LIST OF ABBREVIATIONS**

NN	Neural Network
CNN	Convolutional Neural Network
MNIST	Modified National Institute of Standardand Technology
HTML	Hypertext Markup Language
CSS	Cascading Style Sheet

# CHAPTER 1

## INTRODUCTION

A system and method are provided for automatically estimating a repair cost for a vehicle. A method includes: receiving, at a server computing device over an electronic network, one or more images of a damaged vehicle from a client computing device; performing image processing operations on each of the one or more images to detect external damage to a first set of parts of the vehicle; inferring internal damage to a second set of parts of the vehicle based on the detected external damage; and, calculating an estimated repair cost for the vehicle based on the detected external damage and inferred internal damage based on accessing a parts database that includes repair and labor costs for each part in the first and second sets of parts.

### 1.1 PROJECT OVERVIEW

“Intelligent Vehicle Damage Assessment and Cost Estimator for Insurance Companies” Is the project developed to automate the process of insurance claiming. The rise of AI has been phenomenal in various fields, and the BFSI sector is no different. AI systems can analyze accident images to identify repair costs on a real-time basis. Insurance companies can experience better productivity by implementing AI in insurance claims processing and managing manual time for important tasks. AI accelerates the claims process and helps insurance companies experience better productivity.

Furthermore, in situations like a pandemic where social distancing norms are necessary, surveyors can perform their duties remotely. It not only helps save costs but also leads to savings in time. Automating routine processes like manual damage inspection of cars can overcome inconsistencies that can prove to be costly errors. Document capture technologies also help in handling large volumes of documents at once.

## **1.2 PURPOSE**

By using application or managing expense tracking will help to control unnecessary expenses. It will distribute your expenses in different categories suitable for the user. An expense history will also be provided in application. Our goal is to create an “Intelligent Vehicle Damage Assessment & Cost Estimator for Insurance Companies” where user can be claim insurance through online and the process will be automated.

- To estimate the cost for damaged body parts
- To reduce the man power and time consumption
- To reduce the leakage claims and increase the efficiency

## **CHAPTER 2**

### **LITERATURE SURVEY**

#### **2.1 EXISTING PROBLEM**

Intelligent damage assessment system can assist the damage locator in the front-end damage detection process. The operator only needs to take several photos to upload according to the requirements, and the system can automatically identify the damage degree of the damaged parts and components. The system in the back-end nuclear damage link can provide auxiliary nuclear damage and anti-fraud services. It can identify the cases of fixed-loss errors through the logical recognition of vehicle parts, image fraud recognition, fixed-loss logic recognition, etc. At the same time, it can also meet the demands of anti-fraud and leakage prevention. At present, the intelligent damage assessment system can realize the appearance damage of passenger cars, including CAR, SUV, MPV and VAN. The applicable damage range covers all types of damage of vehicle exterior parts; the applicable environment range covers rain and snow environment, dark environment (vehicle can be seen by human eyes), strong light environment and other scenarios.

#### **2.2 REFERENCES**

Title:

Automatic assessment of damage and repair costs in vehicles

Authors:

Vikas Taliwal, Boston, Siddhartha Dalal, Kaigang Li, Brooklyn

Description:

A method for automatically estimating a repair cost for a vehicle, comprising, receiving, at a server computing device over an electronic network, one or more images of a damaged vehicle from a client computing device; performing image processing operations on each of the one or more images to detect external damage to a first set of parts of the vehicle ; inferring internal damage to a second set of parts of the vehicle based on the detected external damage; and, calculating an estimated repair cost for the vehicle based on the detected external damage and



inferred internal damage based on accessing a parts database that includes repair and labour costs for each part in the first and second sets of parts . Additionally in some embodiments, the server computing device may classify the loss as a total, medium or small loss.

Year: October 12, 2017

Title:

Automatic Car Insurance using Image Analysis

Authors:

Aniket Gupta, Jitesh Chogale, Shashank Shrivastav

Description:

Image analysis methods extract information from an image by using semi-automatic or automatic techniques termed: image understanding, image description, scene analysis, pattern recognition, computer/machine vision etc). Image analysis is different from the various other types of image processing methods, such as the restoration or enhancement in that the end result of image analysis procedures is a numerical output rather than an image or some pictorial output. By analyzing different techniques in literature review we conclude different technologies used to provide solutions for insurance companies, such as Srimal Jayawardena uses 3D model of car and other latest papers use CNN model and categories different types of damages which provide efficient machine learning concepts to predict cost evaluation for damage.

Year: April 5, 2020

Title:

Car Damage Detection using Machine Learning

Authors:

Girish N, Mohammed Aqeel Arshad

Description:

One of the key research topics in computer vision is object detection. On the instance level, it determines the category and position information of the object of interest in the image. RCNN, Fast RCNN, Faster RCNN, and SSD are some of the most popular target detection algorithms. These frameworks, on the other hand, necessitate a large quantity of training data and thus end-to-end detection is not possible. The detection frame's positioning ability is limited, and the gradient disappearance or gradient explosion is common when a feature is extracted as the number of convolution layers grows. For these drawbacks, Author proposed a residual network

(ResNet) that uses the residual module to help the model converge, accelerates neural network training, and integrates it with the Mask RCNN target detection model to achieve object detection and segmentation, significantly enhancing model detection accuracy. Mask RCNN is the first deep learning model that incorporate target identification and segmentation in a single network.

Year:

May 15, 2021

## 2.3 PROBLEM STATEMENT DEFINITION

Mr.Abdul is a 55 years old man. He had a own Car and he worked at basic salary for past 30 Years, In this 30 Years he Faced a problem in Choosing Car Damage and Insurance claim.

- Vimal Rajesh wants to know the better recommendation for insurance claiming.
- He has faced huge losses for a long time.
- This problem is usually faced by lot of Customers.
- Mr. Vimal Rajesh needs to know he result immediately for Insurance claim.

Who does the problem affect?	Persons was claim the assessment amount.
What are the boundaries of the problem?	People who vehicle and facing Issues of Insurance claiming.
When does this issue occur?	The issues occurred in damage part not fully estimation interior part not estimated so issues occur in company side.
Where is the issue occurring?	The issue occurs in Automobile industry interior part damage not fully estimated so issues occur in company side, particularly City side.
Why is it important that we fix the problem?	The required for Automobile industry day by developing so the opposite side accident count also increase accident car owner has claim the actual amount of damage so they consider to fix problem And also customer can see the each and every part

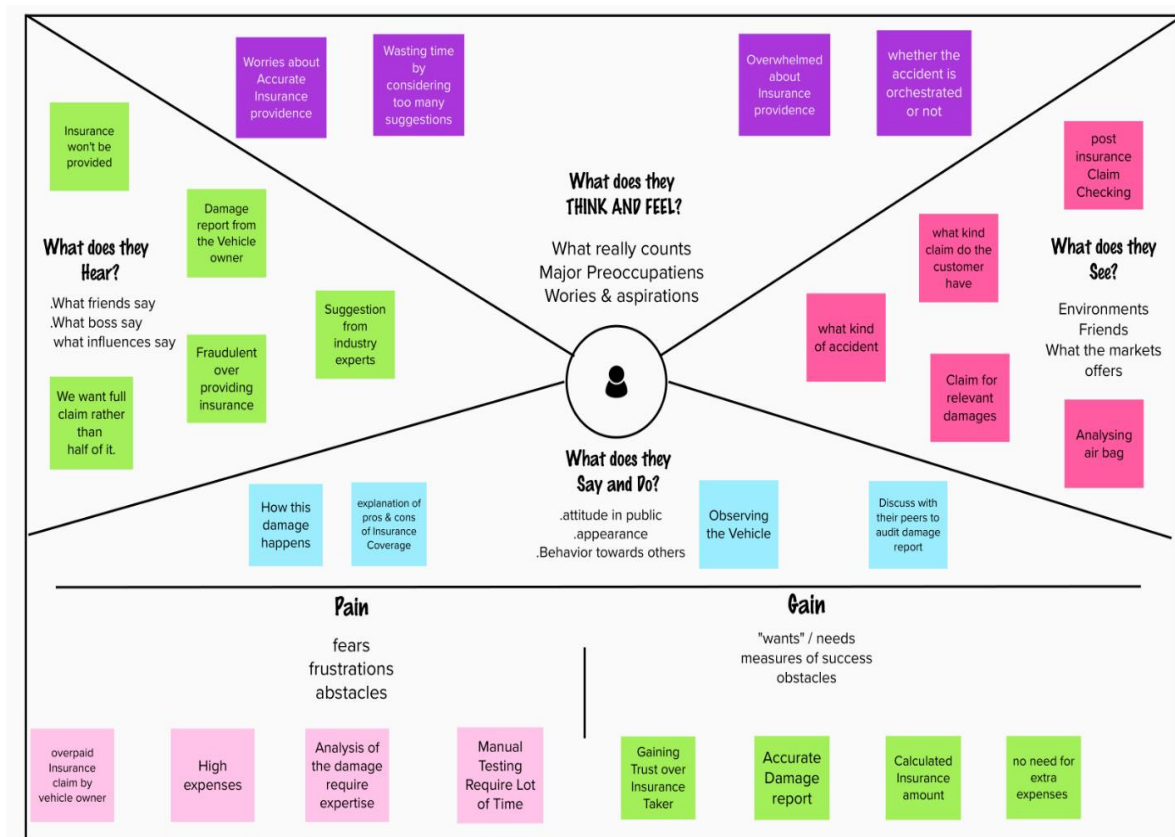
	repairing cost that problem also ratified in this process.
What solution to solve this issue?	An automated system is introduced to identify different diseases on plants by checking the symptoms shown on the leaves of the plant.
What methodology used to solve the issue?	Deep learning techniques are used to identify the specific part of damage repair cost and suggest modify cost rate.

## CHAPTER 3

### IDEATION & PROPOSED SOLUTION

#### 3.1 EMPATHY MAP CANVAS

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user's behaviours and attitudes. It is a useful tool to help teams better understand their users. 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.



## **3.2 IDEATION & BRAINSTORMING**

### **Brainstorm & Idea Prioritization:**

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 number of creative solutions.

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



2

## Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

### TIP

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

SANJAY K

Computer vision based estimation for car

With image preprocessing the user gets notified severity of the damage

VEERAKUMARAVELU M

We can estimate the minor, major and severe damage of the car

With the help of the image we can detect the dent

VGG model achieves almost 92% accuracy in image net

Neural network extracting image feature

Fake image detection

Identifying the problem severity using CNN

NITHESH V

We have to acquire correct information about the damage of car

We must give the user authentication ID

NEETHIARASAN S

With the help of Artificial Intelligence we provide easy estimations about damage

Accuracy and transparency in pricing car and their potential repairs

We should redirect the user to authorized insurance company

Terms and condition should be accessible to any OS

Car model detection

Images are clearly detected using VGG model

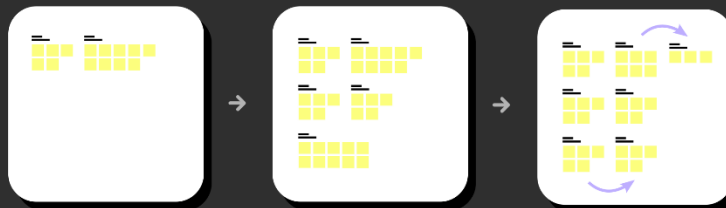
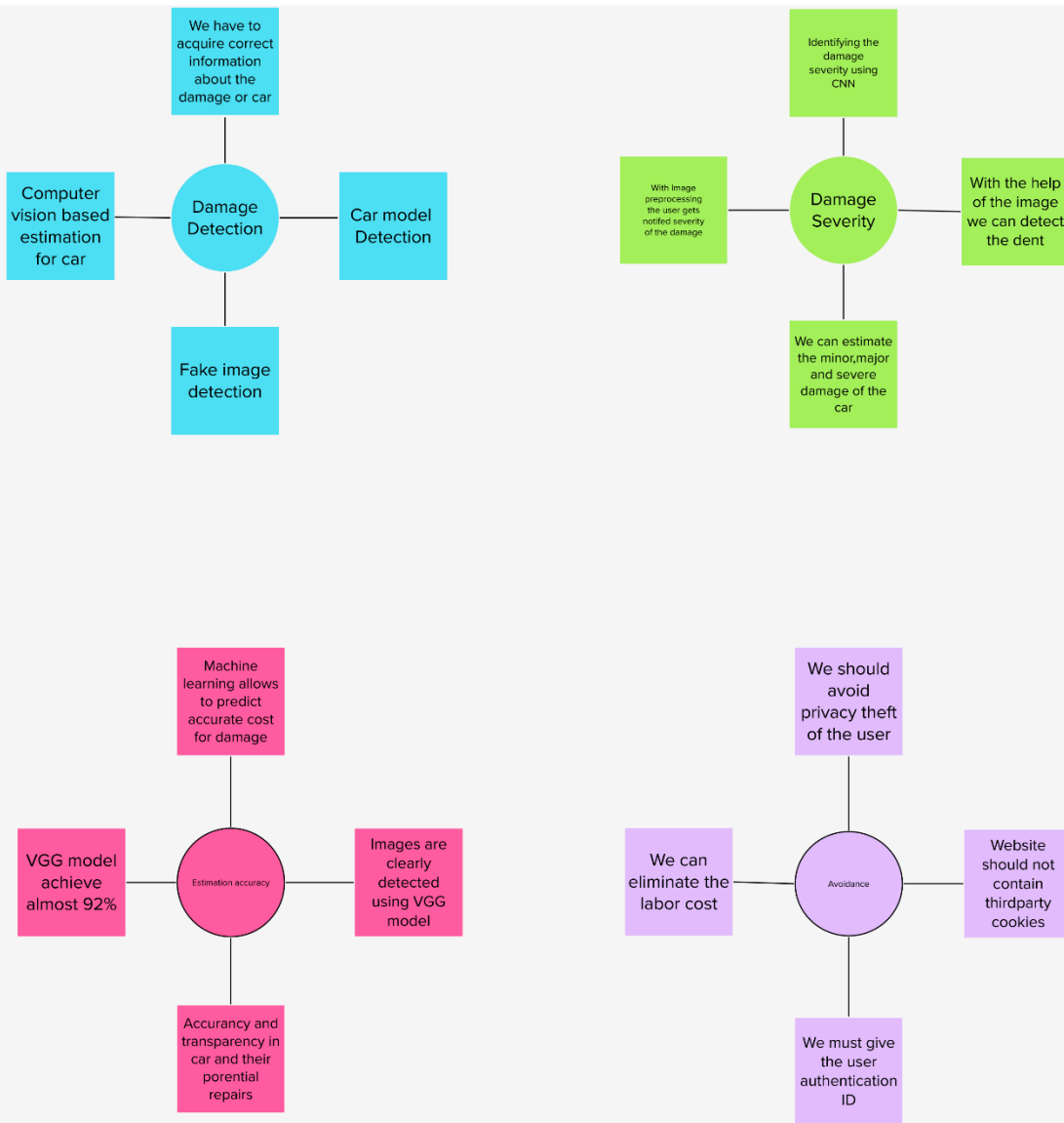


3

## Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

⌚ 20 minutes





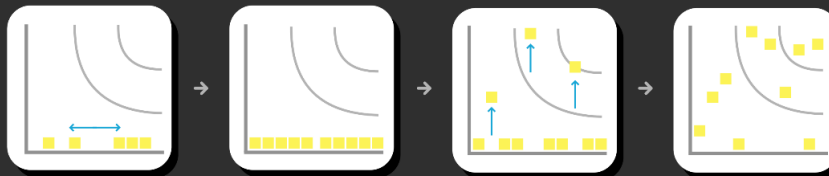
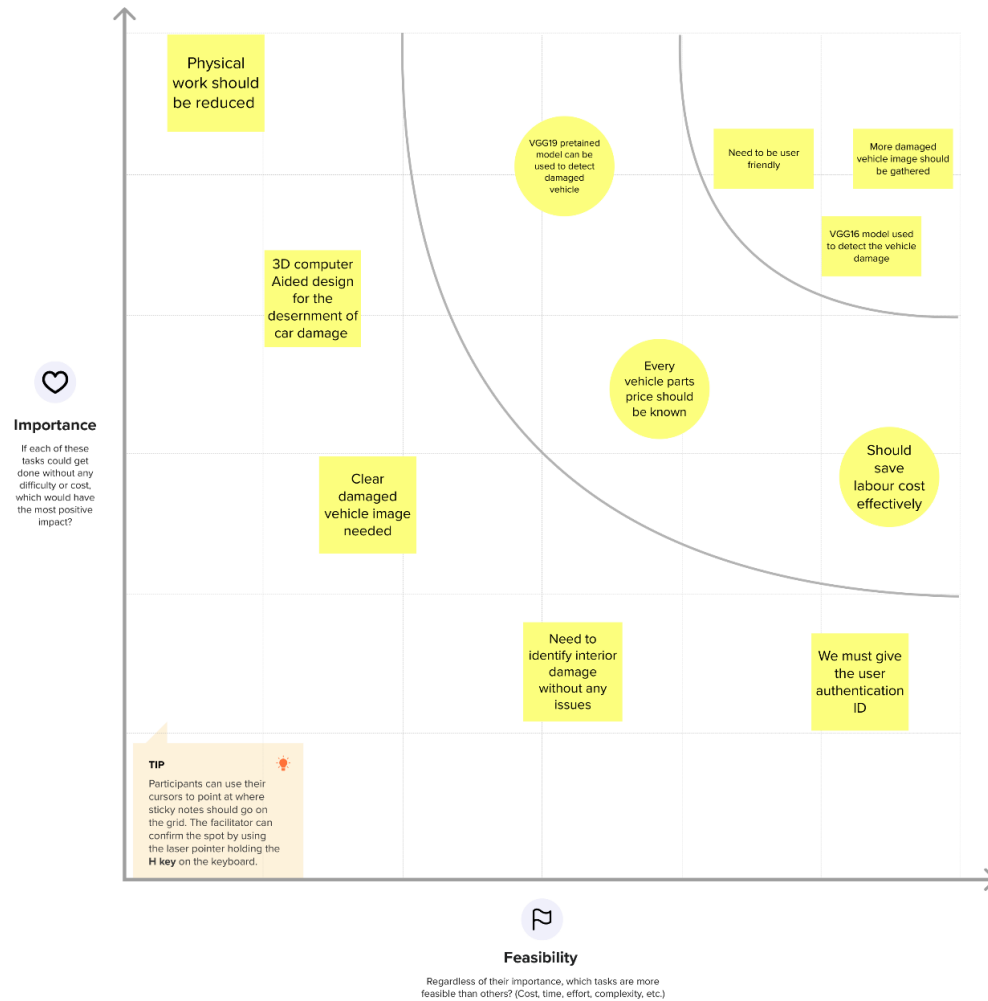
### **Step-3: Idea Prioritization**

4

## Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

🕒 20 minutes



### 3.3 PROPOSED SOLUTION

S.NO	PARAMETER	DESCRIPTION
1	Problem Statement ( Problem to be solved )	Insurance company frequently suffer loses. Because they did not provide a proper explanation regards the estimation of the damage to the customer.
2	Idea / Solution Description	<ol style="list-style-type: none"> <li>1. We create an AI Model to sense and detect the precise amount damage that occurred inthe vehicle.</li> <li>2. Then we create a user accessible portal andsecurely store the dataprovided by the user.</li> <li>3. Finally compare the gathered image percentage with the statistical cost estimation value to predict the cost.</li> </ol>
3	Novelty / Uniqueness	<ol style="list-style-type: none"> <li>1.The AI Model automatically calculatesthe damaged vehicle cost.</li> <li>2.The deep learning algorithm provides progressively higher level features.</li> </ol>

4	Social Impact / Customer Satisfaction	<p>1. It's the user friendly website.</p> <p>2. All the images and personal data will be secured in the cloud data security.</p>
5	Business Model ( Revenue Model )	<p>Insurance companies have two primary sources of income Underwriting &amp; Investment income. Financial investments including Listed shares, Government bonds, and Corporate bonds, make up the majority of insurance firms' assets. By estimating the level of car damage using our AI model and providing insurance accordingly, they can save more money and invest it in their businesses.</p>

6	Scalability of the Solution	<p>1. With the use of advanced machine Learning techniques analyze damaged vehicles with high accuracy levels and keep on improving the learning ability of the model.</p> <p>2. Our AI model can operate at the scale, speed, and complexity required for the aim.</p>
---	-----------------------------	---

## 3.4 PROBLEM SOLUTION FIT

Problem-Solution fit canvas 2.0		Purpose / Vision	
Define CS, fit into	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span> <ul style="list-style-type: none"> <li>Commercial working people traveling from one point to another.</li> <li>Basically belonging to 18+ year's old.</li> <li>Person who's vehicle experienced from accident or damaged in the vehicle.</li> <li>Customer with the valid insurance policy to claim.</li> </ul>	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span> <ul style="list-style-type: none"> <li>The most common constraints faced by the customer is network connection because of the internet availability</li> </ul>	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span> <ul style="list-style-type: none"> <li>Approaching third person for the cost estimation.</li> <li>Cost estimation done by manual calculation.</li> <li>Using slow processing algorithm to detect the damage.</li> </ul> <p>Pros :</p> <ul style="list-style-type: none"> <li>The estimated value stays with in the customer and bank agent.</li> </ul> <p>Cons :</p> <ul style="list-style-type: none"> <li>Estimated cost varies frequently.</li> <li>The time taken for estimation is very high leading to lots of losses and mental issues.</li> </ul>
	<b>Explore AS, differentiate</b>		
Focus on J&P, tap into BE, understand RC	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> <ul style="list-style-type: none"> <li>One of the major problems faced by the customers or the insurance companies are not having idea about the cost of repair for the damage.</li> <li>Insurance companies are failing to provide right amount for the car damage and the customers not able to claim for the damage</li> </ul>	<b>3. PROBLEM ROOT CAUSE</b> <span>RC</span> <ul style="list-style-type: none"> <li>Deviation or variation from the companies calculated cost and the actual cost</li> <li>Rapid development in the AI field paved way too many advanced methodologies of cost estimation.</li> <li>Customer have to do it the cost of the change in regulations.</li> </ul>	<b>7. BEHAVIOUR</b> <span>BE</span> <ul style="list-style-type: none"> <li>The customer has to upload the images of their car after the accident.</li> <li>The application will instantly evaluate the damage and displace the claim amount to the customer.</li> </ul>
	<b>Focus on J&amp;P, tap into BE, understand RC</b>		
Identify strong TR &	<b>3. TRIGGERS</b> <span>TR</span> <ul style="list-style-type: none"> <li>Reading about the more solutions in the news and various websites</li> <li>Development of new technologies.</li> </ul>	<b>10. YOUR SOLUTION</b> <span>SL</span> <ul style="list-style-type: none"> <li>Accurately the estimate the damage percentage.</li> <li>Predict the region of damage with respect to the vehicle.</li> <li>Eliminating human error while estimation.</li> <li>Use of fast processing algorithm for functionality</li> </ul>	<b>1. CHANNELS of BEHAVIOUR</b> <span>CH</span> <p>1. ONLINE</p> <p>Customer Interact with the webpage through internet.</p> <p>2. OFFLINE</p> <p>Customer cannot access this webpage without internet.</p>
	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> <p>Before the customer are not able to claim accurate amount for the damage in vehicle. After the technology development the customer felt independent and comfortable to use the technologies and the solution can be more.</p>	<b>Extract online &amp; offline CH of BE</b>	



## CHAPTER 4

### REQUIREMENT 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	<b>User Registration</b>	Registration through Form Registration through Gmail
FR-2	<b>User Confirmation</b>	Confirmation via Email Confirmation via OTP
FR-3	<b>User details</b>	Users are required to register their personal details likename, age, date of birth, driving license, car number etc.
FR-4	<b>User requirements</b>	The user simply inputs vehicle damage images. The software will instantly generate an accurate reading of 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.

<b>NFR No.</b>	<b>Non-Functional Requirement</b>	<b>Description</b>
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>	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 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	<b>Reliability</b>	This application must perform without failure in 90 percent of use cases during a month. it is more reliable.
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.



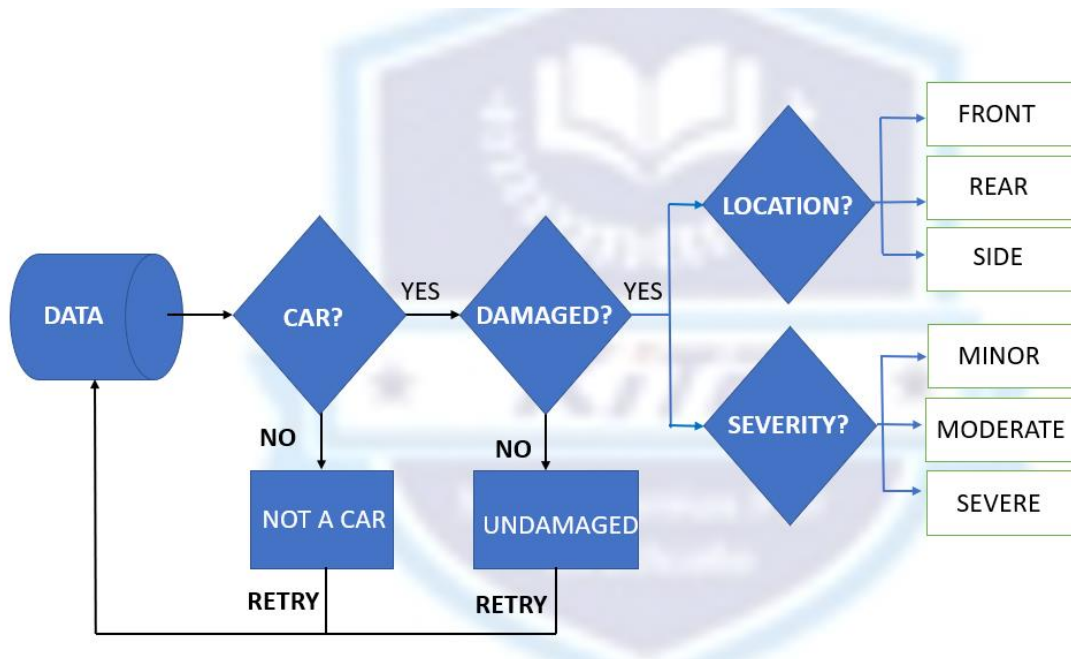
NFR-5	<b>Availability</b>	The web dashboard must be available to user's 99.9 percent of the time every month during business hours EST. Users can access any time and any where.
NFR-6	<b>Scalability</b>	The application must be scalable enough to support 10,000 visits at the same time while maintaining optimal performance and efficient to retrieve image in large scale thus improving scalability.

## CHAPTER 5

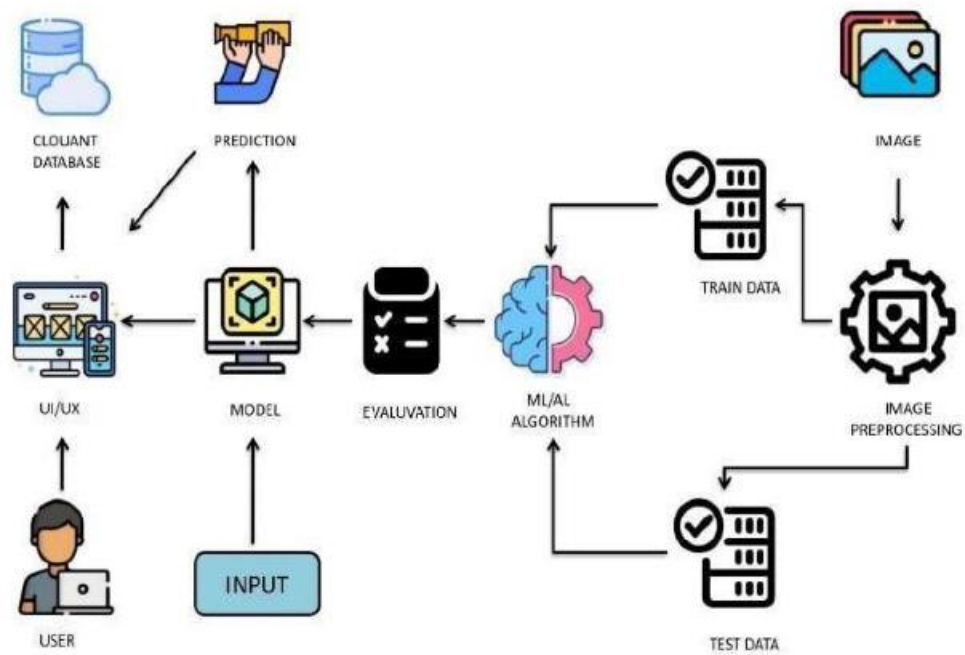
### PROJECT DESIGN

#### 5.1 Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored.



## 5.2 Solution & Technical architecture



## 5.3 User Stories

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	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	High	Sprint-1
	Login	USN-2	As a user, I can register for the application through Gmail	I can access my account/dashboard	High	Sprint-1
	Dashboard	USN-3	As a user, I can log into the application by entering email & password	I get all the info needed in my dashboard.	Low	Sprint-2
	Order creation	USN-4	As a customer, I can place my order with the detailed description of my query	I can ask my query	Medium	Sprint-2
	Forgot password	USN-5	As a customer, I can reset my password by this option in	I get access to my account again	High	Sprint-3

			case I forgot my old password.			
	Order details	USN-6	As a Customer, I can see the current stats of order	I get better understanding	Medium	Sprint-4
Agent (Web user)	Login	USN-1	As an agent I can login to the application by entering correct email and password	I can access my account/dashboard	Medium	Sprint-3
	Dashboard	USN-2	As an agent I can see the order details assigned to me by admin	I can see the tickets to which I could answer	High	Sprint-3
	Address column	USN-3	As an agent I get to have conversation with the customer and clear his/her doubts	I can clarify issues	High	Sprint-3
	Forgot password	USN-4	As an agent I can reset my password by this option In	I get to access to my account again	Medium	Sprint-4

			case I forgot my old password			
Admin user (mobile user and web user)	Login	USN-1	As a admin, I can login to the application by entering email and password	I can access my account/dashboard	High	Sprint-1
	Dashboard	USN-2	As an admin I can see all the orders raised in the entire system and lot more	I can assign agents by seeing those order.	High	Sprint-1
	Agent creation	USN-3	As an admin I can create an agent for clarifying the customers queries	I can create agents	High	Sprint-2
	Assignment agent	USN-4	As an admin I can assign an agent for each order created by the customer	Enable agent to clarify the queries.	High	Sprint-1

	Forgot password	USN- 5	As an admin I can reset my password by this option in case I forgot my old password.	I get access to account	High	Sprint- 1
--	--------------------	-----------	---	----------------------------	------	--------------

## CHAPTER 6

### PROJECT PLANNING & SCHEDULING

#### 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Homepage	USN-1	AS a user I can view the index page to see the about of the Expense tracker	10	High	Sanjay K
Sprint-1	Registration	USN-2	As a User, I need to register user id and passcode for every workers over there in municipality	10	High	Veerakumaravelu M
Sprint-1	Login	USN-3	As a user, I need to login with user id and password to get in to the website	10	High	Neethiarasan S



Sprint-2	Dashboard	USN-4	As a User, I will follow Co-Admin's instruction to reach the filling bin in short roots and save time	20	Low	Nithesh,Neethiarasan
Sprint-3	Uploading image	USN-5	As a User I will add my expense throughout the month I spend on	20	Medium	Veerakumaravelu M
Sprint-3	Cost Estimation	USN-6	As a User I can view my expense in a graph of overview of the expense I spend.	20	Medium	Sanjay,veerakumaravelu
Sprint-4	Deployment in cloud	USN-7	As a User I can access the cloud to store my data of expense	20	High	Neethiarasan, Veerakumaravelu, Sanjay,nithesh

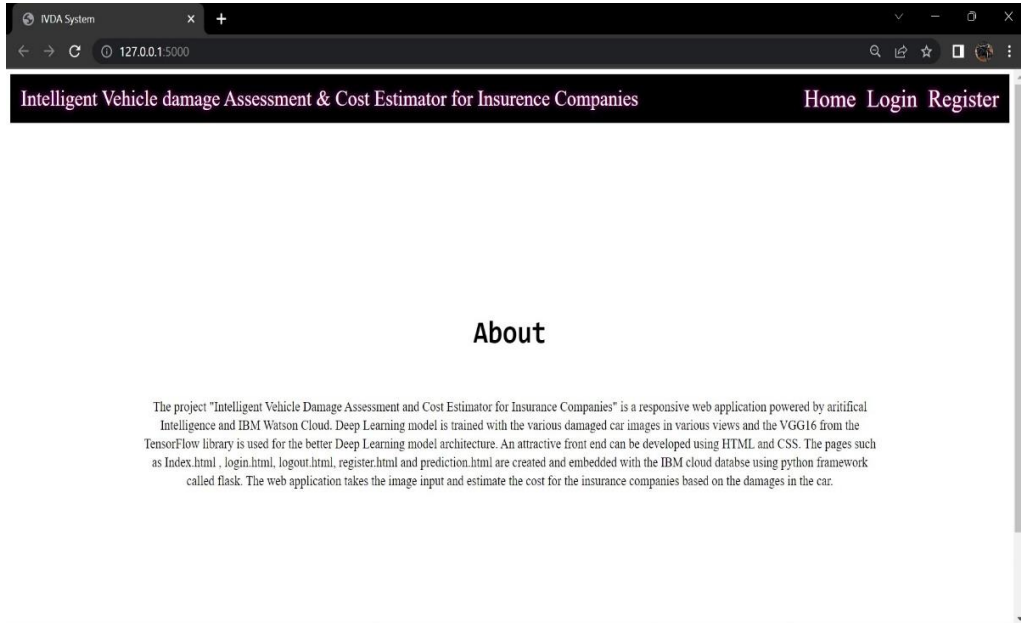
## 6.2 Sprint Delivery Schedule

<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-1	20	6 Days	23 Oct 2022	28 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	30 Oct 2022	04 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	06 Nov 2022	11 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	13 Nov 2022	18 Nov 2022	20	19 Nov 2022

## CHAPTER 7

### CODING AND SOLUTIONING

#### 7.1 FEATURE 1



#### 7.2 FEATURE 2



The Estimated Cost of the Damage:9000 - 11000 INR

## CHAPTER 8

### TESTING

#### 8.1 TEST CASES

TEST CASE ID	FEATURE TYPE	COMPONENT	TEST SCENARIO	EXPECTED RESULT	ACTUAL RESULT	STATUS
TC_001	UI	Home Page	Verify UI elements in the Home Page	The Home Page must be displayed properly	Working as expected	PASS
TC_002	UI	Home Page	Verify whether the page is responsive	The Home Page must display in the same way in all devices	The UI is displayed correctly only on the desktop screens	FAIL
TC_003	Functional	Home Page	Check if user could navigate to the next page	The button in the Home Page is directing to next page	Working as expected	PASS
TC_004	Functional	Backend	Check if all the routes are working properly	All the routes should properly work	Working as expected	PASS
TC_005	Functional	Model	Check if the model can handle various image sizes	The model should rescale the image and predict the results	Working as expected	PASS
TC_006	Functional	Model	Check if the model predicts the amount	The model should predict the Amount	Working as expected	PASS

TC_007	Functional	Prediction Page	Reports error if files are not uploaded	Prediction Page pops out error page if file is not uploaded	Working as expected	PASS
TC_008	UI	Prediction Page	Verify UI elements in the Prediction Page	The Prediction page must be displayed properly	Working as expected	PASS
TC_009	UI	Prediction Page	Check if the result is displayed properly	The result should be displayed properly	Working as expected	PASS
TC_010	Functional	Model	Check if the model can handle Complex input image	The model should predict the amount in the complex image	The model fails to identify the amount since the model is not built to handle such data	FAIL

## 8.2 USER ACCEPTANCE TESTING

Acceptance Testing is a level of the software testing where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery. Formal testing with respect to user needs, requirements, and business processes conducted to determine whether or not a system satisfies the acceptance criteria and to enable the user, customers or other authorized entity to determine whether or not to accept the system. In this application, the customer's acceptance is been monitored and it is been put into usage.

### 8.2.1 TEST CASE ANALYSIS

SECTION	TOTAL CASES	NOT TESTED	FAIL	PASS
Client Application	5	0	1	4
Security	1	0	0	1
Performance	3	0	1	2
Exception Reporting	1	0	0	1

# CHAPTER 9

## RESULTS

### 9.1 PERFORMANCE METRICS

Script: locust.py

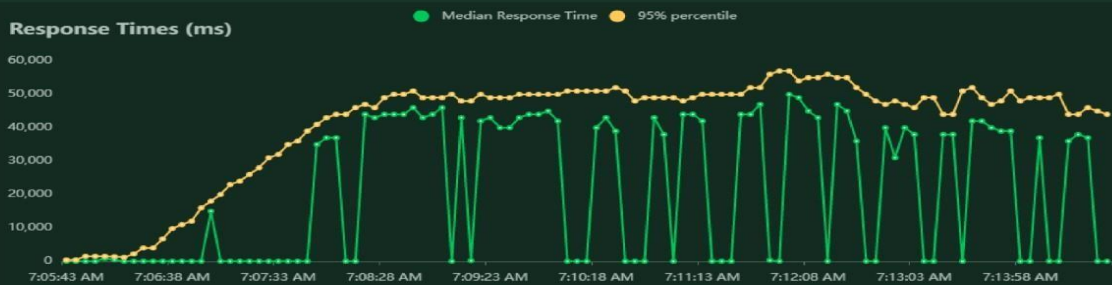
#### Request Statistics

Method	Name	# Requests	# Fails	Average (ms)	Min (ms)	Max (ms)	Average size (bytes)	RPS	Failures/s
GET	//	1043	0	13	4	290	1079	1.9	0.0
GET	//predict	1005	0	39648	385	59814	2670	1.8	0.0
Aggregated		2048	0	19462	4	59814	1859	3.7	0.0

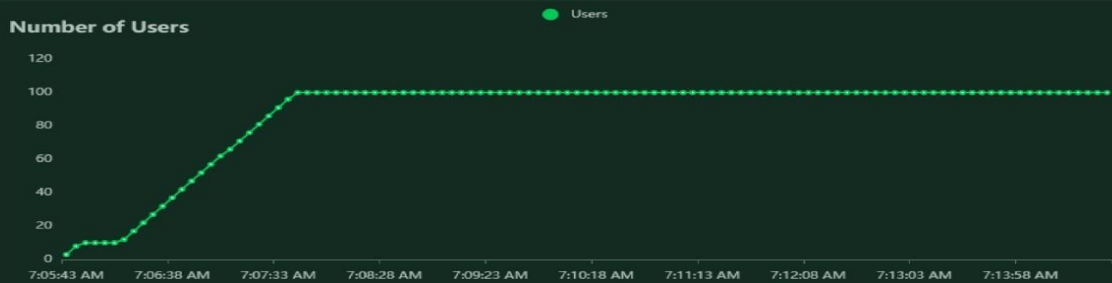
#### Response Time Statistics

Method	Name	50%ile (ms)	60%ile (ms)	70%ile (ms)	80%ile (ms)	90%ile (ms)	95%ile (ms)	99%ile (ms)	100%ile (ms)
GET	//	10	11	13	15	19	22	62	290
GET	//predict	44000	46000	47000	48000	50000	52000	55000	60000
Aggregated		36	36000	43000	45000	48000	50000	54000	60000

#### Response Times (ms)



#### Number of Users



## **CHAPTER 10**

### **ADVANTAGES AND DISADVANTAGES**

#### **10.1 ADVANTAGES**

- Estimating damages with over 90% accuracy: Our model for vehicle damage detection is trained with tens of thousands of real images of car accidents. Providing over 90% accuracy for the damage estimation and cost estimation.
- Speeding up the claims process : It is quite evident that using artificial intelligence solutions that can automatically approximate the cost of damages in minutes, would cut down on the extra time taken in the traditional claims process. Hence, insurance holders will not need to wait for weeks for their claim to be approved. This contributes to greatly improved customer experiences.
- Elimination of mistakes: Everything is subject to human error and AI solutions can help prevent inaccuracies in the estimation of damages and costs, which can save a lot of money. There is no need to worry about mistakes being made regarding the estimation of damage caused to a vehicle in an accident because the AI software does all the work in an efficient and reliable manner.
- Versatility: The application can be used for any vehicle, irrespective of type, model, origin, etc. This makes the solution versatile as it can be used on any vehicle. In an accident, the model and the type of vehicle heavily influence the amount of money needed for damage reparation. AI can be used to estimate these costs in no time, as it can recognize the unique characteristics of each vehicle and act accordingly.
- Severity analysis: It is of paramount importance to know how severe the damages are. A map of the areas affected by the accident is drawn out and the severity is rated. This provides a user-friendly solution to understand how each part of the vehicle has been affected by the accident.



## **DISADVANTAGES**

- Manually tracking all cash that is spent can be irritating as well as time consuming
- In the growth of technology, images are processed more by the computer vision in a large amount. Together with the trend, there is also the simultaneous rise of fraudulent claims in insurance as a deceptive act. The claims can be repeated for identical damages in order to deceive the system or people, which leads to the big loss to the insurance carrier

## **CHAPTER 11**

### **CONCLUSION**

In the future, we will continue to explore the innovation of insurance technology of 'AI + Vehicle Insurance'. We hope that we can use the power of intelligent damage determination system. On the one hand, the owner can take photos by one click to achieve rapid loss determination, price estimation and immediate compensation. On the other hand, it assists insurance companies to achieve rapid and accurate pricing in the process of fixing losses and claims. Finally, by combining the rapid compensation of accident vehicles to relieve traffic pressure, to avoid more serious personal and property losses caused by secondary accidents.

## **CHAPTER 12**

### **FUTURE SCOPE**

In the future, application can be improved with following features:

- Display the image of the car with scanned analysis.
- Multiple cars can be scanned at a time.
- Now, only cars can be analyzed. In future every vehicles will be made to be scanned and analyzed.
- To build an android app to support on mobile platforms.

## CHAPTER 13

### APPENDIX

#### 13.1 SOURCE CODE

##### App.py

```
from cloudant.client import Cloudant
import os
import tensorflow
from keras.utils import load_img, img_to_array
from werkzeug.utils import secure_filename
import numpy as np
from keras.models import load_model
from tensorflow.python.ops.gen_array_ops import concat
from keras.applications.inception_v3 import preprocess_input

#creating the Cloudant Database
client = Cloudant.iam("4d6e563a-8f74-40cf-959d-a767edfb9e6f-
bluemix","KsH9257a5n7jNU3aWE1zvCKn-1-
wSFw47DVt4_R5zKFm",connect=True)
database = client.create_database("bath4_database")

#load model
model1 = load_model('C:\\Users\\DELL\\Desktop\\IBM PROJECT 4\\IBM-
Project-23426-1659882722\\Final Deliverables\\model\\body.h5')
model2 = load_model('C:\\Users\\DELL\\Desktop\\IBM PROJECT 4\\IBM-
Project-23426-1659882722\\Final Deliverables\\model\\level.h5')
```

```
from flask import Flask,render_template,request,redirect,url_for
```

```
app = Flask(__name__)
```

```
@app.route('/')
```

```
def home():
```

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

```
#login page setting
```

```
@app.route('/login')
```

```
def login():
```

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

```
@app.route('/afterLogin',methods=['POST','GET'])
```

```
def afterlogin():
```

```
    user = request.form['_id']
```

```
    passw = request.form['psw']
```

```
    print(user,passw)
```

```
    query = {'_id':{'$eq':user}}
```

```
    docs = database.get_query_result(query)
```

```
    print(docs)
```

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

```
    if(len(docs.all())==0):
```

```

        return render_template('login.html',message='The username is not found')
    else:
        if((user==docs[0][0]['_id'] and passw==docs[0][0]['psw'])):
            return redirect(url_for('prediction'))
        else:
            return render_template("login.html",message="Invalid User Details")

#Register page setting

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

@app.route('/afterRegister',methods=['POST'])
def afterregister():
    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 = database.get_query_result(query)

```

```

if(len(docs.all())==0):
    url = database.create_document(data)
    return render_template('register.html', message="Registration is
Successfully Completed")
else:
    return render_template("register.html", message="You are already a
member!")

```

### **prediction.Html**

```

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

```

#logout page

```

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

```

#results

```

@app.route('/result', methods = ['GET', 'POST'])
def upload_file():
    if request.method == 'POST':
        f = request.files['_file']

```

```
basepath = os.path.dirname(__name__)  
filepath = os.path.join(basepath,'uploads',f.filename)  
f.save(filepath)
```

```
img = load_img(filepath,target_size=(224,224))  
x = 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','near','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=="near" and result2=="minor"):  
    value="4000 to 6000 Inr"  
elif(result1=="near" and result2=="moderate"):
```



```

        value="7000 - 9000 Inr"
    elif(result1=="near" 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 - 11000Inr"
    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=True)

```

### **Index.html**

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
        rel="stylesheet"

```

```

href="{ { url_for('static', filename = 'style.css') } }"
/>
<title>IVA System</title>
</head>
<body>
<nav class="navbar">
  <div class="navbar__heading">
    Intelligent Vehicle damage Assessment & Cost Estimator for Insurance
Companies
    </marquee>
  </div>
  <div class="navbar__items">
    <a class="navbar__items__text" href="/">Home</a>
    <a class="navbar__items__text" href="/login">Login</a>
    <a class="navbar__items__text" href="/register">Register</a>
  </div>
</nav>

<main>
  <div class="main__content">
    <h1 class="main__content__heading">About</h1>
    <p class="main__content__para">
      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.

</p>

</div>

</main>

<footer class="footer">

<div class="footer\_\_text">©Copyright@2022</div>

<div class="footer\_\_logo">

<a href="https://www.linkedin.com/in/neethiarasan-s-37a023194/"

><svg

xmlns="http://www.w3.org/2000/svg"

width="18"

height="18"

fill="#fff"

class="bi bi-linkedin"

viewBox="0 0 16 16"

>

<path

d="M0 1.146C0 .513.526 0 1.175 0h13.65C15.474 0 16 .513 16 1.146v13.708c0 .633-.526 1.146-1.175 1.146H1.175C.526 16 0 15.487 0 14.854V1.146zm4.943 12.248V6.169H2.542v7.225h2.401zm-1.2-8.212c.837 0 1.358-.554 1.358-1.248-.015-.709-.52-1.248-1.342-1.248-.822 0-1.359.54-1.359

1.248 0 .694.521 1.248 1.327 1.248h.016zm4.908 8.212V9.359c0-.216.016-.432.08-.586.173-.431.568-.878 1.232-.878.869 0 1.216.662 1.216 1.634v3.865h2.401V9.25c0-2.22-1.184-3.252-2.764-3.252-1.274 0-1.845.7-2.165 1.193v.025h-.016a5.54 5.54 0 0 1 .016-.025V6.169h-2.4c.03.678 0 7.225 0 7.225h2.4z"/> </svg> </a>

<a href="https://github.com/neethiarasan30"><svg  
xmlns="http://www.w3.org/2000/svg"  
width="18"  
height="18"  
fill="#fff"  
class="bi bi-github"  
viewBox="0 0 16 16">  
<path  
d="M8 0C3.58 0 0 3.58 0 8c0 3.54 2.29 6.53 5.47 7.59.4.07.55-.17.55-.38 0-.19-.01-.82-.01-1.49-2.01-.37-2.53-.49-2.69-.94-.09-.23-.48-.94-.82-1.13-.28-.15-.68-.52-.01-.53.63-.01 1.08.58 1.23.82.72 1.21 1.87.87 2.33.66.07-.52.28-.87.51-1.07-1.78-.2-3.64-.89-3.64-3.95 0-.87.31-1.59.82-2.15-.08-.2-.36-1.02.08-2.12 0 0 .67-.21 2.2.82.64-.18 1.32-.27 2-.27.68 0 1.36.09 2 .27 1.53-1.04 2.2-.82 2.2-.82.44 1.1.16 1.92.08 2.12.51.56.82 1.27.82 2.15 0 3.07-1.87 3.75-3.65 3.95.29.25.54.73.54 1.48 0 1.07-.01 1.93-.01 2.2 0 .21.15.46.55.38A8.012 8.012 0 0 0 16 8c0-4.42-3.58-8-8-8z"/></svg></a>

</div>

</footer>

</body>

</html>

## Login.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="{{ url_for('static', filename = 'style.css') }}"
    />
    <title>Login Page</title>
    <style>
      body{
        background-color: rgba(255, 0, 0, 0);
      }
      form {
        margin: 0 auto;
        margin-top: 20px;
      }
      label {
        color: #555;
        display: inline-block;
        margin-left: 18px;
        padding-top: 10px;
        font-size: 14px;
      }
```

```

input[type=text],
input[type=password] ,input[type=time]{
color: #777;
padding-left: 10px;
margin: 10px;
margin-top: 12px;
margin-left: 18px;
width: 290px;
height: 35px;
border: 1px solid #c7d0d2;
border-radius: 2px;
box-shadow: inset 0 1.5px 3px rgba(190, 190, 190, .4), 0 0 0 5px #f5f7f8;
-webkit-transition: all .4s ease;
-moz-transition: all .4s ease;
transition: all .4s ease;
}
</style>
</head>
<body>

```

```

<nav class="navbar">
  <div class="navbar__heading">Login Page</div>
  <div class="navbar__items">
    <a class="navbar__items__text" href="/">Home</a>
    <a class="navbar__items__text" href="/login">Login</a>
  
```

```

    <a class="navbar__items__text" href="/register">Register</a>
</nav>
<register_container class="_register_">
    <form action="afterLogin" method="post">
        <label for="Email">Enter your Email ID</label>
        <input type="text" id="_id" name="_id">
        <label for="password">Enter the Password</label>
        <input type="password" id="psw" name="psw">
        <div>
            <input type="submit" value="Login">
        </div>
    </form>
    <div>
        <h3>{{ message }}</h3>
    </div>
</register_container>
</body>
</html>

```

## Logout.html

```

<!DOCTYPE html>
<html lang="en">
    <head>
        <meta charset="UTF-8" />
        <meta http-equiv="X-UA-Compatible" content="IE=edge" />
        <meta name="viewport" content="width=device-width, initial-scale=1.0" />
        <link
            rel="stylesheet"

```

```

    href="{{ url_for('static', filename = 'style.css') }}"
  />
<title>Login Page</title>
</head>
<body>
  <nav class="navbar">
    <div class="navbar__heading">Logout Page</div>
    <div class="navbar__items">
      <a class="navbar__items__text" href="/">Home</a>
      <a class="navbar__items__text" href="/login">Login</a>
      <a class="navbar__items__text" href="/register">Register</a>
    </div>
  </nav>
  <main>
    <div class="main__content">
      <h1 class="main__content__heading">Successfully Logged Out!</h1>
    </div>
  </main>
</body>
</html>

```

### **Prediction.html**

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link

```



```

    rel="stylesheet"
    href="{ { url_for('static', filename = 'style.css') } }"
/>
<title>prediction</title>
</head>
<body>
    <nav class="navbar">
        <div class="navbar__heading">Vehicle Damage Detection</div>
        <div class="navbar__items">
            <a class="navbar__items__text" href="/">Home</a>
            <a class="navbar__items__text" href="/logout">Logout</a>
        </div>
    </nav>
    <form action="result" method="post" enctype="multipart/form-data">
        Select image to upload:
        <input type="file" name="_file" id="_file">
        <input type="submit" value="submit" name="submit">
    </form>
    <main>
        <div class="main__content">
            <h1 class="main__content__heading">The Estimated Cost of the
Damage:{ {prediction} }</h1>
        </div>
    </main>
</body>
</html>

```

## Register.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link
      rel="stylesheet"
      href="{ { url_for('static', filename = 'style.css') } }"
    />
    <title>Registration Page</title>
    <style>
      body {
        background-color: rgba(255, 0, 0, 0);
      }
      form {
        margin: 0 auto;
        margin-top: 20px;
      }
      label {
        color: rgb(0, 0, 0);
        display: inline-block;
        margin-left: 18px;
        padding-top: 10px;
        font-size: 14px;
      }
```

```

input[type=text],
input[type=password] ,input[type=time]{
color: #777;
padding-left: 10px;
margin: 10px;
margin-top: 12px;
margin-left: 18px;
width: 290px;
height: 35px;
border: 1px solid #c7d0d2;
border-radius: 2px;
box-shadow: inset 0 1.5px 3px rgba(190, 190, 190, .4), 0 0 0 5px #f5f7f8;
-webkit-transition: all .4s ease;
-moz-transition: all .4s ease;
transition: all .4s ease;
}

```

```

</style>

```

```

</head>

```

```

<body>

```

```

<nav class="navbar">

```

```

<div class="navbar__heading">Vehicle Damage Detection</div>

```

```

<div class="navbar__items">

```

```

<a class="navbar__items__text" href="/">Home</a>

```

```

<a class="navbar__items__text" href="/login">Login</a>

```

```

<a class="navbar__items__text" href="/register">Register</a>

```

```

</nav>

```

```

<register_container class="_register_">
  <form action="afterRegister" method="post">
    <label for="Name">Enter Your Name</label>
    <input type="text" id="Name" name="name" placeholder="Your
name..">
    <label for="Email">Enter your Email ID</label>
    <input type="text" id="Email" name="Email" placeholder="Your mail
ID..">
    <label for="password">Enter the Password</label>
    <input type="password" id="password" name="password"
placeholder="Password..">
    <div>
      <input type="submit" value="Register" >
    </div>
  </form>
  <div>
    <h3>
      {{ message }}
    </h3>
  </div>
</register_container>
</body>
</html>

```

## Style.css

```
@import url("://cdn.jsdelivr.net/npm/bootstrap-icons@1.9.1/font/bootstrap-  
icons.css");
```

```
https
```

```
*{  
    margin: 0;  
    padding: 0;  
    box-sizing: border-box;  
}
```

```
.navbar{  
    display: flex;  
    justify-content: space-between;  
    align-items: center;  
    padding: 16px 20px;  
    background-color: rgb(0, 0, 0);  
    color: #fff;  
}
```

```
.navbar__heading {  
    font-size: 25px;  
    text-shadow: 0 0 10px #FF0000, 0 0 10px #0000FF;;  
}
```

```
.navbar__items {  
    display: flex;
```

```
gap: 20px;
align-items:center;
text-shadow: 0 0 10px #FF0000, 0 0 10px #0000FF;;

}
```

```
.navbar__items__text {
  cursor: pointer;
  text-decoration: none;
  color:rgb(255, 255, 255);
  font-size: 25px;

}
```

```
.main__content {
  min-height: 90vh;
  display: flex;
  justify-content: center; /* Remove later */
  flex-direction: column;
  align-items: center;
  padding: 50px;
  background-color: rgba(255, 0, 0, 0);

}
```

```
.main__content__heading {
```

```
font-size: 50px;
text-decoration: solid;
}
```

```
.main__content__para {
padding: 20px 200px;
text-align: center;
font-size: 20px;
line-height: 30px;
font-size: 22px;
}
```

```
.footer {
display: flex;
justify-content: space-between;
align-items: center;
padding: 16px 20px;
background-color: #000;
color: #fff;
}
```

```
.footer__text{
font-size: 20px;
}
```

```
.footer__logo {
display: flex;
```

```

    gap: 20px;
    align-items: center;
    color: #fff;
}

._login_{
    position: fixed;
    width: 340px;
    height: 175px;
    top: 50%;
    left: 50%;
    margin-top: -140px;
    margin-left: -170px;
    background: #fff;
    border-radius: 3px;
    border: 1px solid #ccc;
    box-shadow: 0 1px 2px rgba(0, 0, 0, .1);
}

._register_{
    position: fixed;
    width: 340px;
    height: 280px;
    top: 50%;
    left: 50%;
    margin-top: -140px;
    margin-left: -170px;
    background: rgb(255, 255, 255);
    border-radius: 3px;

```



## **13.2 GitHub & Project Demo Link:**

**13.2.1 GITHUB LINK - <https://github.com/IBM-EPBL/IBM-Project-46173-1660740470>**

**13.2.2 Project Demo Link - <https://drive.google.com/drive/u/0/folders/1MYsnEsEerLustnz8rT5k4GhkJdNzYmfL>**

## CHAPTER 14

### REFERENCES

- [1] LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D. Backpropagation applied to handwritten zip code recognition. Neural computation, 1989, pp. 541-551.
- [2] Krizhevsky, A., Sutskever, I., Hinton, G. Imagenet classification with deep convolutional neural networks. In NIPS, 2012, pp. 1097-1105.
- [3] Zeiler, M. D., Fergus, R. Visualizing and understanding convolutional neural networks. In ECCV, 2014, pp. 818-833.
- [4] LeCun, Y., Bengio, Y., Hinton, G. Deep learning. Nature, 2015(521), pp. 436-444.
- [5] Simonyan, K., Zisserman, A. Very deep convolutional networks for large-scale image recognition. In ICLR, 2015, pp. 1409.1556.
- [6] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In NIPS, 2015, pp. 91-99.
- [7] Kaiming He, Georgia Gkioxari, Piotr Dollar, Ross Girshick. Mask R-CNN. In ICCV, 2017, pp. 2980-2988