Project On

Intelligent Vehicle Damage Assessment & Cost Estimator for Insurance Companies

powered By IBM India

Submitted By
Solai Sri Varshini R
Sri Ranjini N
Sruthi V
Vidhyasri R

Project ID: PNT2022TMID16533

of
Computer Science And Engineering
S.A Engineering College
Poonamallee, Avadi main road , Chennai - 600077

College Mentor: **Gunanandhini**
Industrial Mentor: **Swathi**

INDEX

1. INTRODUCTION

1.1. Project Overview

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

1.2. Purpose

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

2. LITERATURE SURVEY

2.1. Existing Problem

The problem is defined as the optimzed way to estimate insurance cost based on the mulitple damages in the various areas in the vechile for the insurance companies. As the existing methods for estimating the cost takes lot of time and energy in the way of inspecting the vechile.

2.2. References

S.No	Author/ Publication Year	Title	Methodology	Advantages	Disadvantages
1.	Phyu Mar Kyu, Kuntpong Woraratpanya, 2021	Car Damage Assessment Based on VGG Models.	CNN models on ImageNet dataset to perform different tasks of localization and detection. YOLO object detection model to train and detect damage region as their important pipeline to improve their performance of damage detection.	Transfer learning and regularization can work better than those of fine tuning. Pretrained models assess its location and security which help insurance companies to solve claim leakage problems.	CNN cannot accurately calculate the level of damage part. Sometimes overfitting occurs.

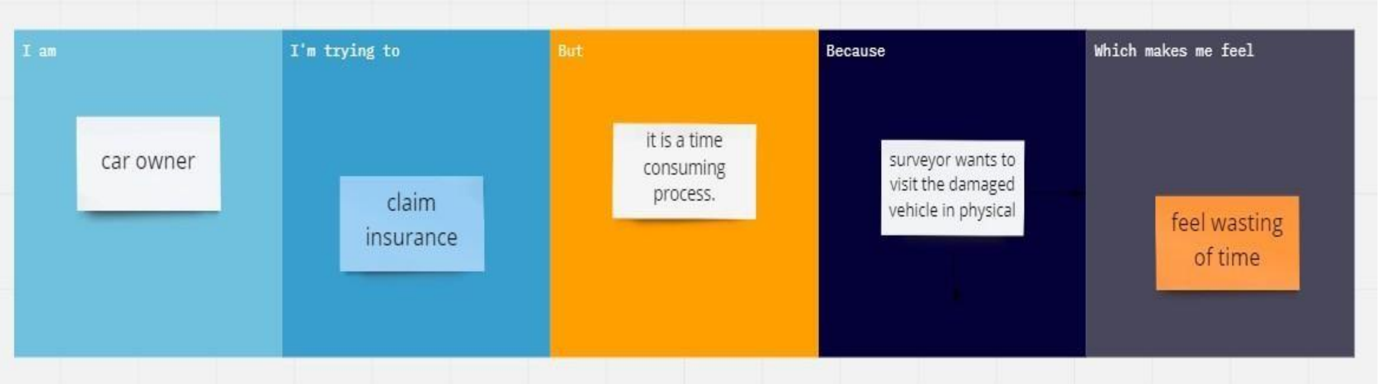
2.	Najmeddine Dhieb, Hakim Ghazzai, Hichem Besbes Yehia Massoud, 2019	A Very Deep Transfer Learning Model for Vehicle Damage Detection and Localization.	Combination of deep learning, instance segmentation, and transfer learning techniques for features extraction and damage identification.	Transfer learning significantly reduce the training times when it uses the weight of pretrained VGG models. It has progress on how to classify problems when the small dataset was not enough to train a CNN model.	A traditional CNN model can be very time consuming to perform image classification tasks and identify the correct weights for the network by multiple forward and backward iterations.
3.	U. Waqas, N. Akram, S. Kim, D. Lee and J. Jeon, t , 2012	Damage Assessment of a vehicle and Insurance Reclaim.	Deep learning techniques, Moire effect Detection, Mobile Net model is proposed with transfer learning for classification.	It is a hybrid approach which provide only authentic images to algorithm for damage classification as input. moiré effect detection and metadata analysis are performed to detect fraudulent images	The main drawback was that Images in bad lighting, awkward angles, and vehicle models in a small dataset to achieve automation is difficult but still the range is broad.

4.	Li Ying & Dorai Chitra, 2012	Applying image analysis to auto insurance Triage	Image analysis and pattern recognition are applied to automatically identify and characterize automobile damage.	Because of the advancement of image analysis and pattern recognition technologies, the auto insurance industry could significantly benefit.	The drawback is that the automobile damaged can be analyzed only having white background otherwise it will be not able to give the desired results.
5.	Srimal Jayewardene’, 2013	Image based automatic vehicle damage detection	This approach requires 3D computer aided design (CAD) modes of the considered vehicle to identify how it would look if it were undamaged .	Automatically detecting the damage of the vehicle using photographs clicked at the accident site is extremely functional as it can greatly decrease the rate of processing insurance claims, and it will also provide greater conveniences for customers who are making the best use of this functionality.	Vehicles have very reflective metallic bodies the photographs taken in such an uncontrolled environment can be expected to have a certain amount of inter object reflection. Application of standard computer vision techniques is a very challenging task

6.	Phyu Mar Kyu,Kuntpong Woraratpanya ,2020	Car damage detection and classification	CNN model is trained on ImageNet dataset. After fine tuning the dataset, transfer learning with L2 regularization is applied	Pre-trained VGG model not only detect damaged part of a car but also assess its location and severity.	Transfer learning and regularization can work better than those of fine tuning.
7.	M.Wassel, 2019	A Secure AI- driven Architecture for Automated Insurance Systems: FraudDetecti on and Risk Measurement	Blockchain, data analysis, machine learning, AI for damage identification.	Proposed classifiers ensure not only the best accuracy in detecting fraudulent claims but also can classify different types of fraud for insurance unlike the existing solutions.	The major drawback of the proposed model is that it only identifies the physical visible damage and not of the internal or the interior damage.

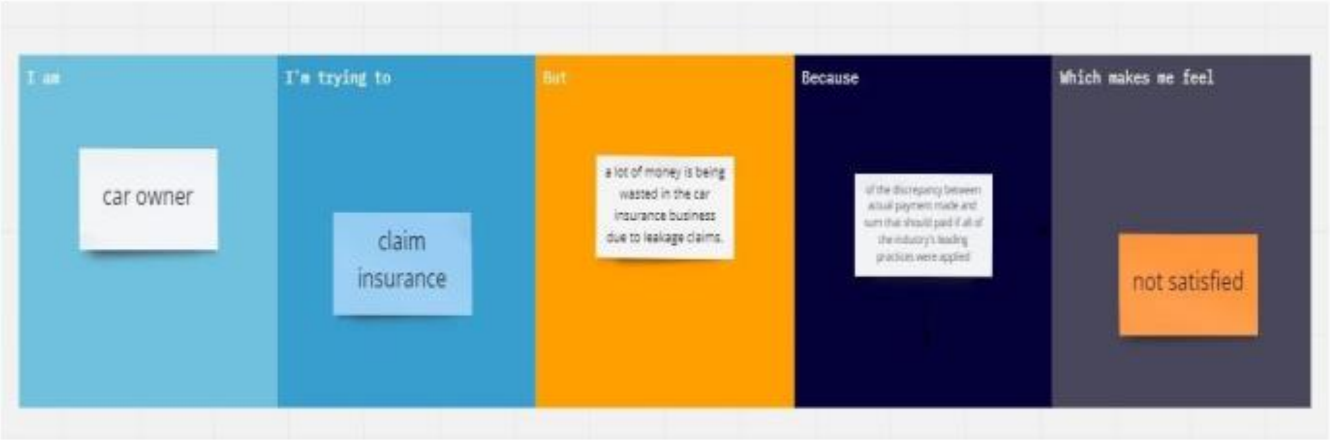
2.3. Problem Statement Definition

Problem Statement - 1



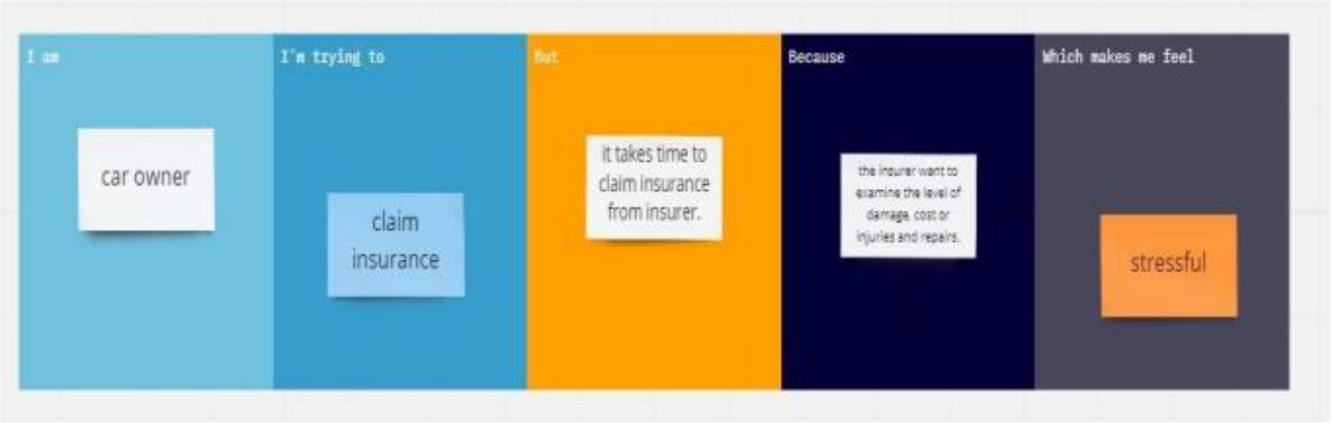
I am	I'm trying to	But	Because	Which makes me feel
car owner	claim insurance	it is a time consuming process.	surveyor wants to visit the damaged vehicle in physical	feel wasting of time

Problem Statement - 2



I am	I'm trying to	But	Because	Which makes me feel
car owner	claim insurance	a lot of money is being wasted in the car insurance business due to leakage claims.	of the discrepancy between actual payment made and sum that should paid if all of the industry's leading practices were applied	not satisfied

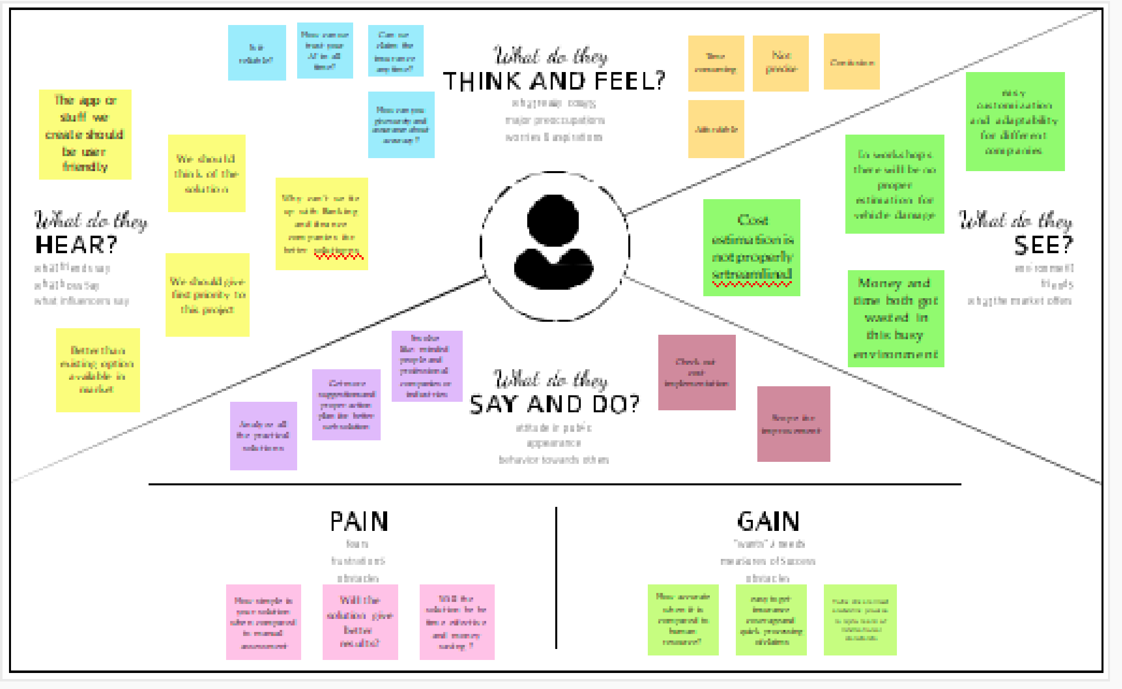
Problem Statement - 3



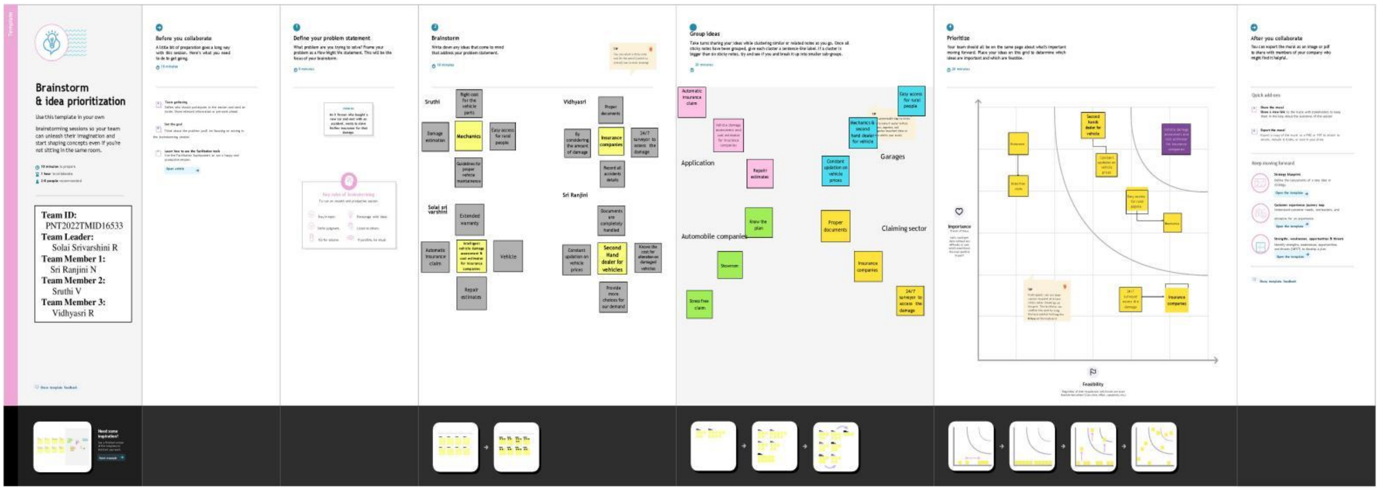
I am	I'm trying to	But	Because	Which makes me feel
car owner	claim insurance	it takes time to claim insurance from insurer.	the insurer want to examine the level of damage, cost or injuries and repairs.	stressful

3.Ideation and Proposed Solution

3.1. Empathy Map


What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Is it reliable?
How can we trust your AI in all times?
Can we claim the insurance any time?
How can you guarantee and assurance about accuracy?
Time consuming
Not precise
Confusion
Affordable
What do they HEAR?
what friends say
what boss say
what influencers say
The app or stuff we create should be user friendly
We should think of the solution
Why can't we tie up with Banking and insurance companies for better solutions
We should give first priority to this project
Better than existing option available in market
What do they SEE?
environment
friends
what the market offers
easy customization and adaptability for different companies
In workshops there will be no proper estimation for vehicle damage
Cost estimation is not properly streamlined
Money and time both got wasted in this busy environment
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
Analyse all the practical solutions
Get more suggestions and proper action plan for better solution
Involve like minded people and professional companies or industries
Check out cost implementation
Scope for improvement
PAIN
fears
frustrations
obstacles
How simple is your solution when compared to manual assessment
Will the solution give better results?
Will the solution be be time effective and money saving ?
GAIN
"wants" / needs
measures of success
obstacles
How accurate when it is compared to human resource?
easy to get insurance coverage and quick processing of claims

3.2. Ideation and Brainstorming



Team ID:
PNT2022TMID16533
Team Leader:
Solai Srivarshini R
Team Member 1:
Sri Ranjini N
Team Member 2:
Sruthi V
Team Member 3:
Vidhyasri R

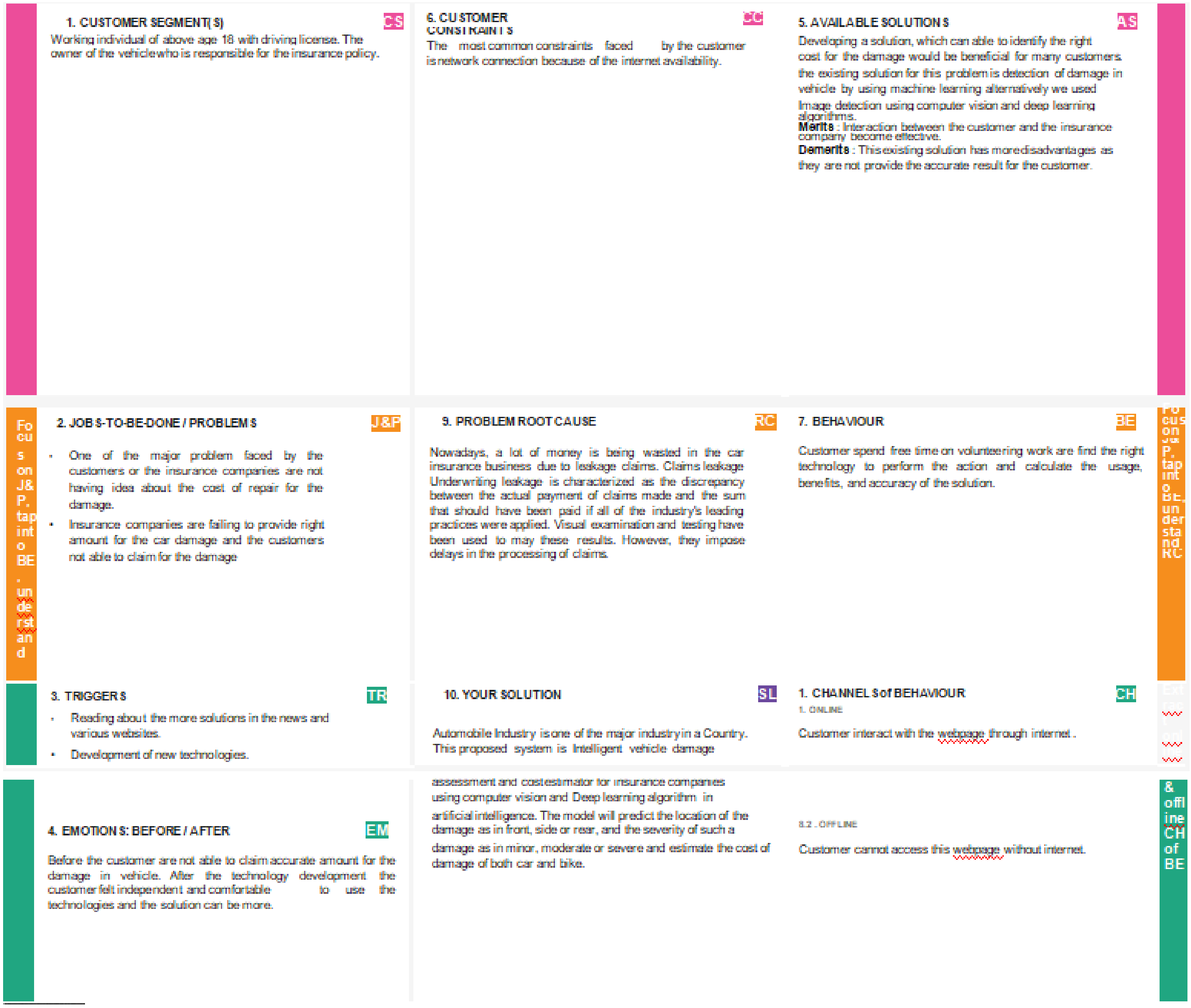
3.3. Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Every asset has a value attached to it that is primarily economic in nature. There is always a risk of these assets being destroyed due to incidents beyond human control. They also may not work due to such events. Depending on the asset class, the type and weight of risk also vary.This is where insurance policies are useful. The problem that might arise is that the claimant maynot know the amount of coverage that he/she has.
2.	Idea / Solution description	1. To develop an optimized and accuratedeep learning architecture to detect the damage percentage and location ofthe damage with respect to the vehicle 2. Implementing classification algorithms to classify damaged regions and implementing the model in web based application 3. Create a user accessible portal and securely store the data provided bythe user 4. Compare the obtained damage percentage with the statistical cost estimation value to predict the cost.

3.	Novelty / Uniqueness	1. The deep learning algorithm will analyze images in real time and identifies the presence of any damage. 2. Even in the presence of minute damages, artificial intelligence can detect the dents and marks on the car’s body. 3. With a lot of training, Artificial intelligence will able to distinguish simple stain from ascratch and effectively estimate the respective damage cost
4.	Social Impact / Customer Satisfaction	1. All the features of this project will bemade easily accessible to the customers. 2. The webapp is intuitive, easy to use, simple and that the customer can rely on the product. It is easy to start with the appand understand how to use it, high complexity is not valuable for the user. 3. All the uploaded images will be and the personal information of the customerwill be secured in cloud data security. 4. The cost estimation for damages that the webapp provides to the customer will be legitimate and exactto what a normal insurance company offers.

5.	Business Model (Revenue Model)	1. The business model will be a freemium model providing the prediction of damage intensity which will be useful for the vehicle owners to keep track oftheir vehicle damage and the credentials toaccess the webpage can be provided on the purchase of thevehicle insurance. 2. The add-on subscription model can be initiated forthe user where the damagecost is evaluated and provided to the users. 3. The further revenue can be generated by tying upwith the automobile parts manufacturers and distributors by promoting their products to the vehicle that has specified parts damaged.
6.	Scalability of the Solution	1. The damage detection can be providedto all the insured clients to reach the stable base andthen extend the service of cost estimation to the insurers. 2. Make use of advanced machine learning techniquesto analyze the damaged vehicle with high accuracy levels and keep on improving the learning ability ofthe model. 3. In addition to the webpage a mobile application canbe created where the real time images and videos ofthe vehicle canbe extracted and insurance cost can be estimated.

3.4. Proposed Solution Fit



1. CUSTOMER SEGMENT(S) CS

Working individual of above age 18 with driving license. The owner of the vehicle who is responsible for the insurance policy.

6. CUSTOMER CONSTRAINTS CC

The most common constraints faced by the customer is network connection because of the internet availability.

5. AVAILABLE SOLUTIONS AS

Developing a solution, which can able to identify the right cost for the damage would be beneficial for many customers. the existing solution for this problem is detection of damage in vehicle by using machine learning alternatively we used Image detection using computer vision and deep learning algorithms.

Merits : Interaction between the customer and the insurance company become effective.

Demerits : This existing solution has more disadvantages as they are not provide the accurate result for the customer.

2. JOBS-TO-BE-DONE / PROBLEMS J&P

- One of the major problem faced by the customers or the insurance companies are not having idea about the cost of repair for the damage.
- Insurance companies are failing to provide right amount for the car damage and the customers not able to claim for the damage

9. PROBLEM ROOT CAUSE RC

Nowadays, a lot of money is being wasted in the car insurance business due to leakage claims. Claims leakage Underwriting leakage is characterized as the discrepancy between the actual payment of claims made and the sum that should have been paid if all of the industry's leading practices were applied. Visual examination and testing have been used to may these results. However, they impose delays in the processing of claims.

7. BEHAVIOUR BE

Customer spend free time on volunteering work are find the right technology to perform the action and calculate the usage, benefits, and accuracy of the solution.

3. TRIGGERS TR

- Reading about the more solutions in the news and various websites.
- Development of new technologies.

10. YOUR SOLUTION SL

Automobile Industry is one of the major industry in a Country. This proposed system is Intelligent vehicle damage assessment and cost estimator for insurance companies using computer vision and Deep learning algorithm in artificial intelligence. 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 and estimate the cost of damage of both car and bike.

1. CHANNELS of BEHAVIOUR CH

1. ONLINE

Customer interact with the webpage through internet..

8.2 . OFFLINE

Customer cannot access this webpage without internet.

4. EMOTIONS: BEFORE / AFTER EM

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.

Focus on J&P, tap into BE, understand
Focus on J&P, tap into BE, understand RC
Extract online
& offline CH of BE

4. REQUIREMENT ANALYSIS

4.1. Functional Requirement

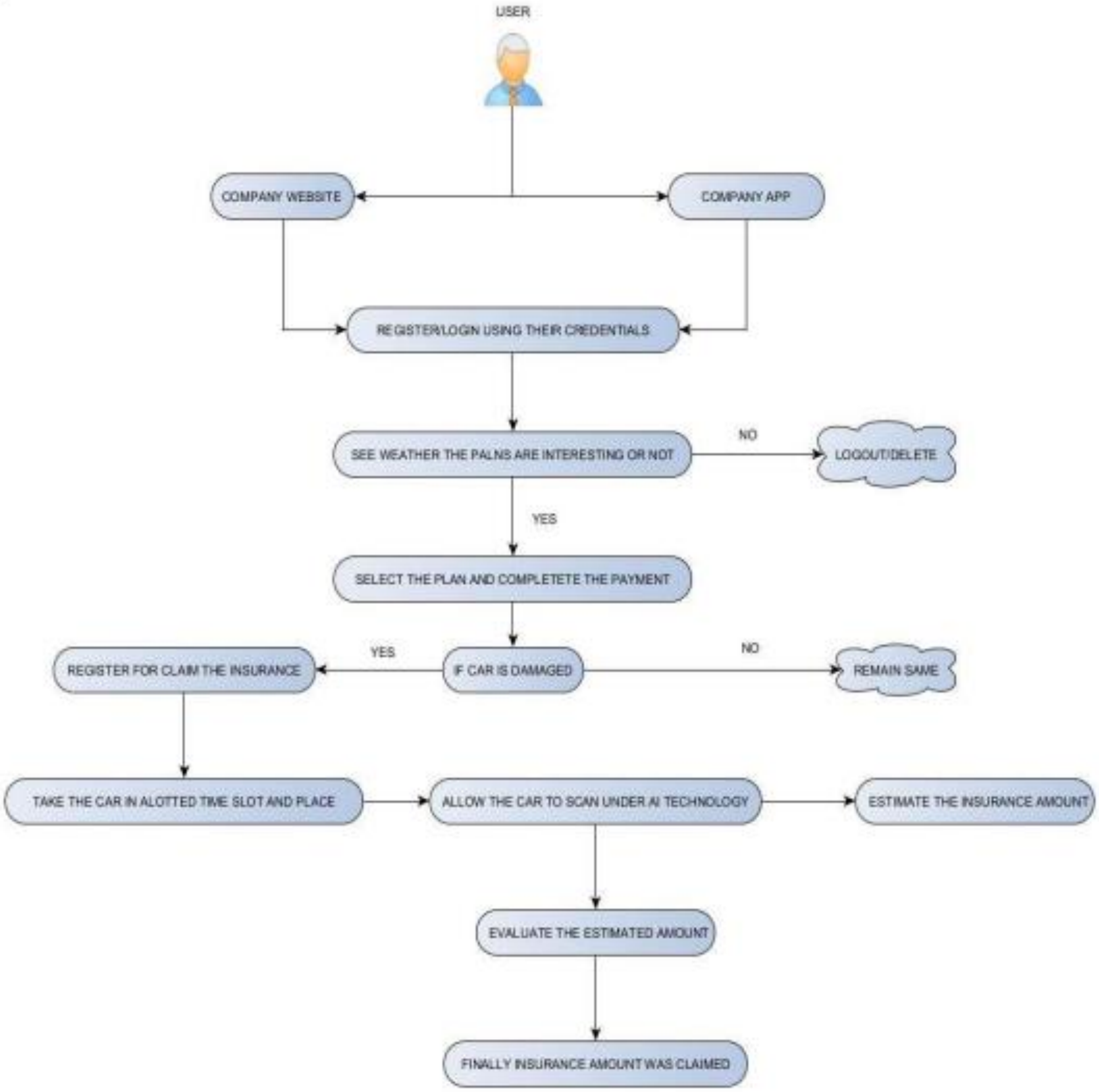
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	Login via Gmail Login via credentials
FR-4	Purchase	Purchase the insurance plan from list
FR-5	Apply	Apply when car got damaged
FR-6	Allocation	Allot a place and time to user for Scanning and estimatethe insurance amount
FR-7	Result	The estimated amount was showed and provided to user

4.2. Non - Functional Requirements

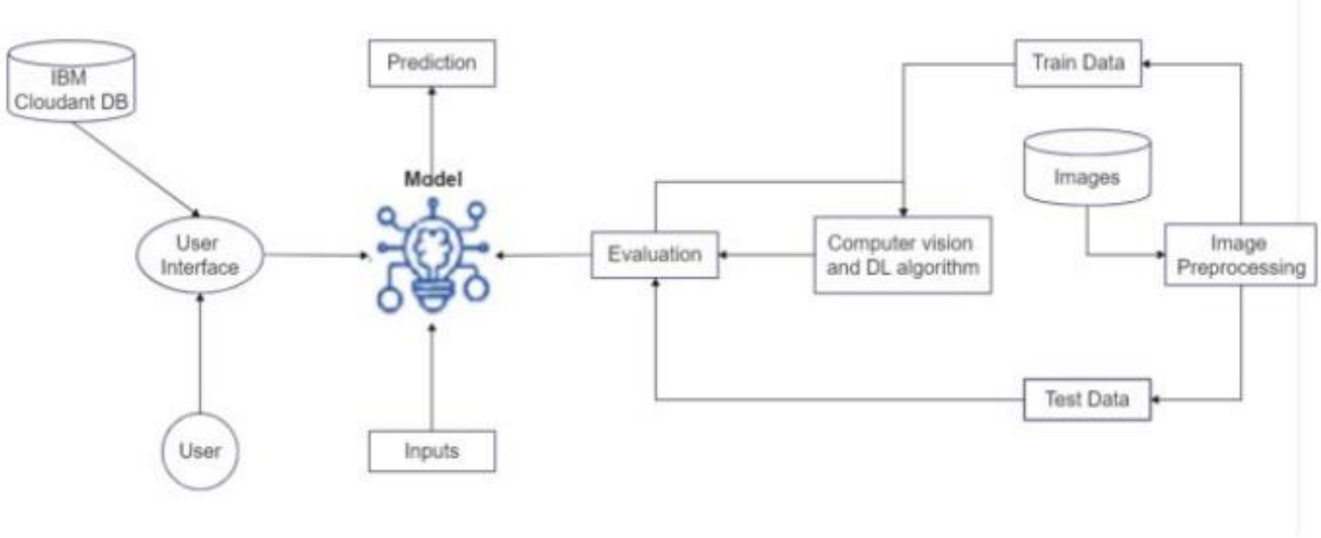
FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	The Final data should be easily understandable .
NFR-2	**Security**	The model are designed in a secured manner inorder to maintain the privacy
NFR-3	**Reliability**	Even if there is a firmware issues (failures) the lastupdated Data's are stored in a Default manner.
NFR-4	**Performance**	High quality sensors are used to ease the customer'swork.
NFR-5	**Availability**	The model are designed in such a way that areavailable ,usable and can be modified anytime.
NFR-6	**Scalability**	The size of the system didn't varies it is made tousable for all types of cars

5. PROJECT DESIGN

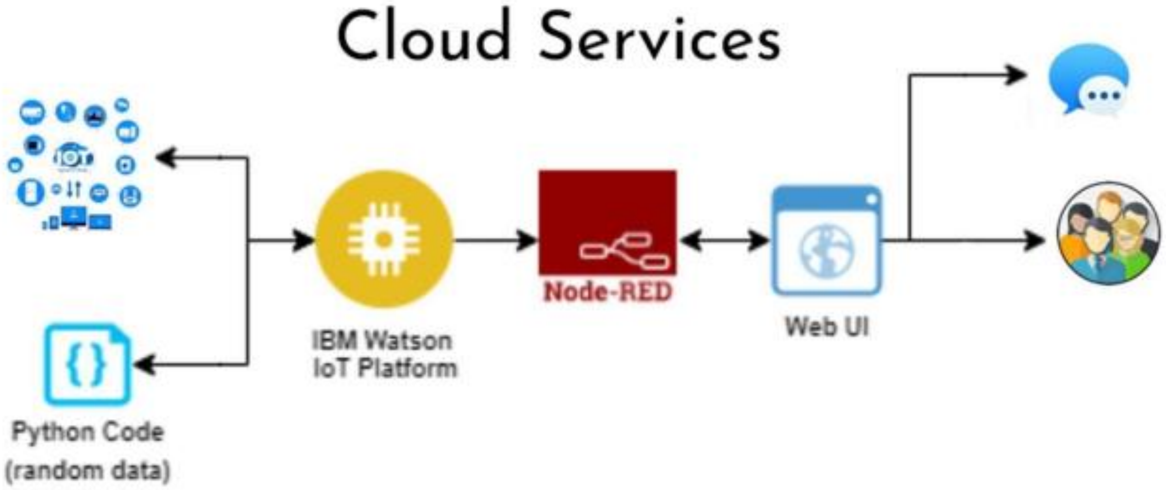
5.1. Data Flow Diagram


USER
COMPANY WEBSITE
COMPANY APP
RE GISTER/LOGIN USING THEIR CREDENTIALS
SEE WEATHER THE PALNS ARE INTERESTING OR NOT
NO
LOGOUT/DELETE
YES
SELECT THE PLAN AND COMPLETETE THE PAYMENT
YES
REGISTER FOR CLAIM THE INSURANCE
IF CAR IS DAMAGED
NO
REMAIN SAME
TAKE THE CAR IN ALOTTED TIME SLOT AND PLACE
ALLOW THE CAR TO SCAN UNDER AI TECHNOLOGY
ESTIMATE THE INSURANCE AMOUNT
EVALUATE THE ESTIMATED AMOUNT
FINALLY INSURANCE AMOUNT WAS CLAIMED

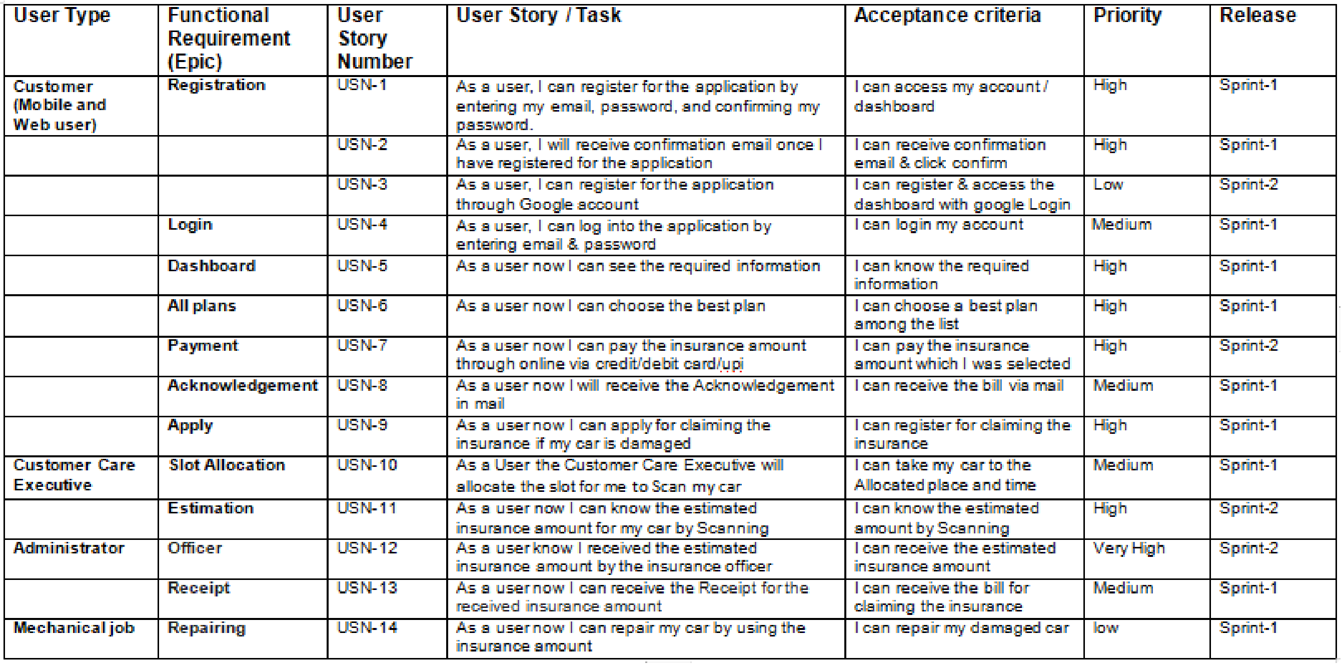
5.2. Solution and Technical Architecture



Technical Architecture



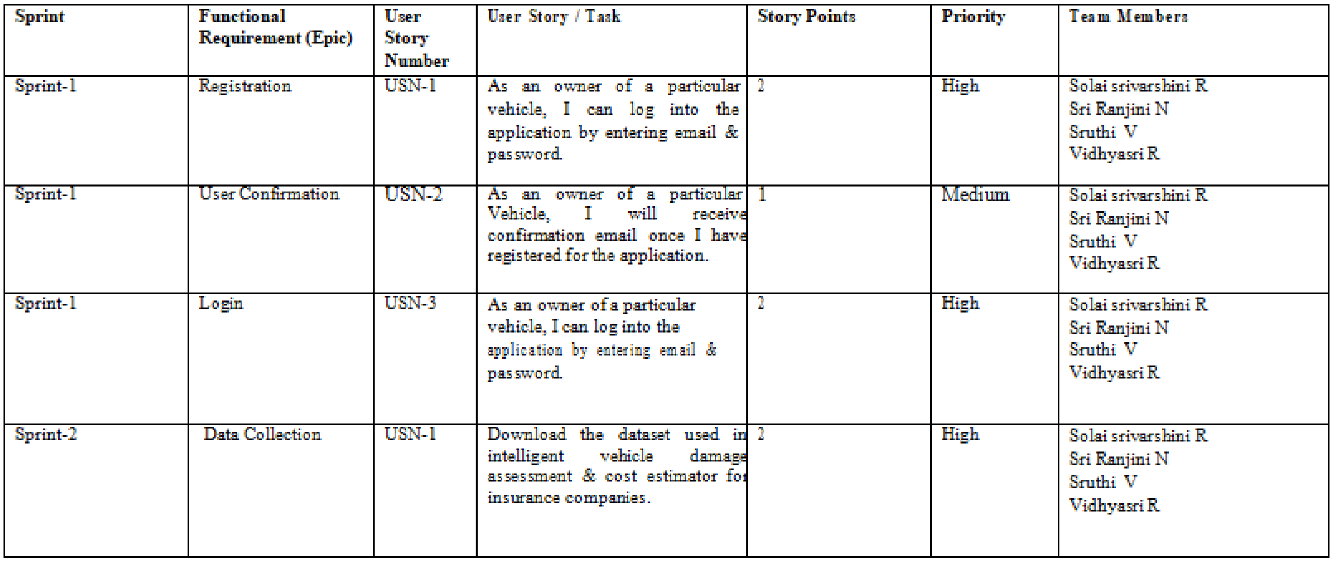
5.3. User Stories



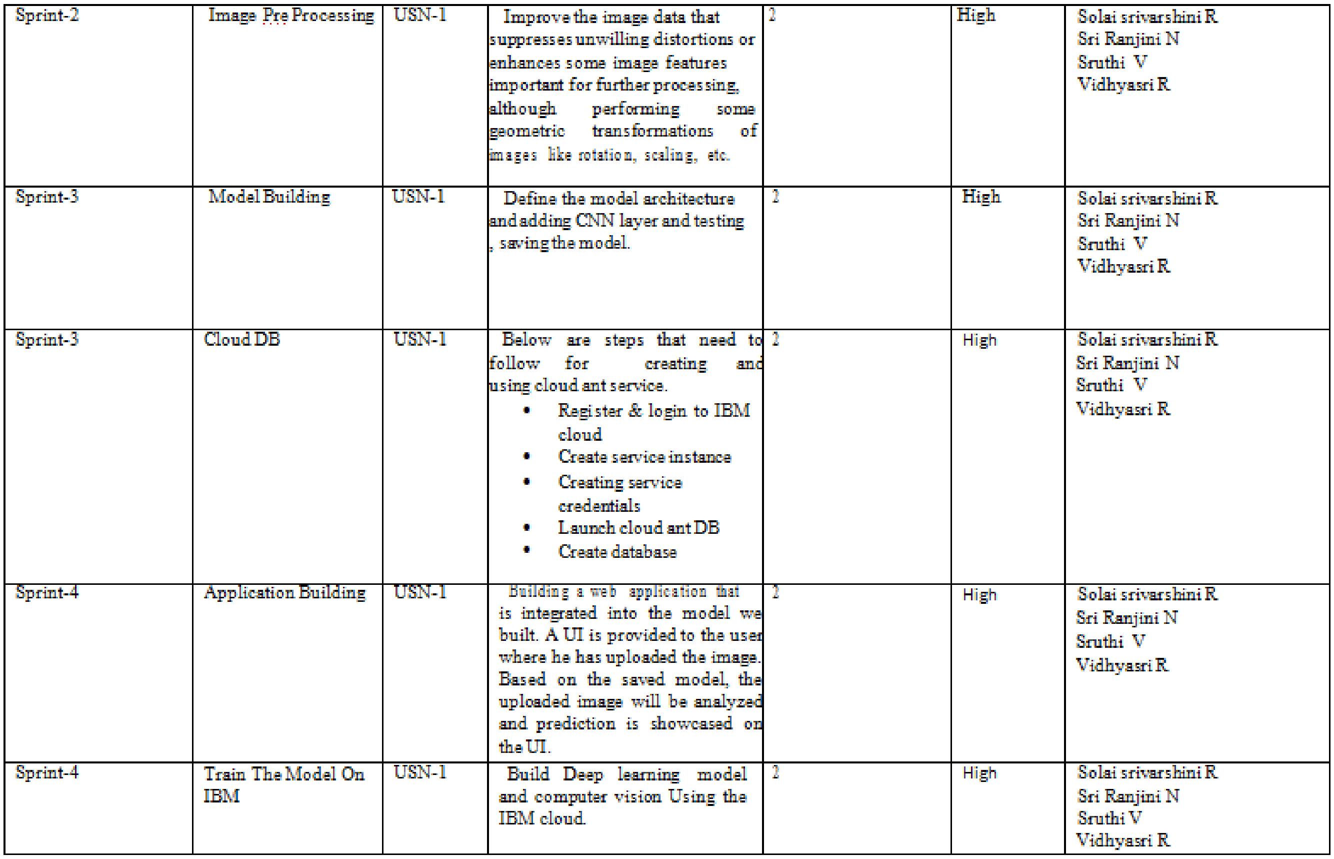
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile and Web user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		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
		USN-3	As a user, I can register for the application through Google account	I can register & access the dashboard with google Login	Low	Sprint-2
	Login	USN-4	As a user, I can log into the application by entering email & password	I can login my account	Medium	Sprint-1
	Dashboard	USN-5	As a user now I can see the required information	I can know the required information	High	Sprint-1
	All plans	USN-6	As a user now I can choose the best plan	I can choose a best plan among the list	High	Sprint-1
	Payment	USN-7	As a user now I can pay the insurance amount through online via credit/debit card/upi	I can pay the insurance amount which I was selected	High	Sprint-2
	Acknowledgement	USN-8	As a user now I will receive the Acknowledgement in mail	I can receive the bill via mail	Medium	Sprint-1
	Apply	USN-9	As a user now I can apply for claiming the insurance if my car is damaged	I can register for claiming the insurance	High	Sprint-1
Customer Care Executive	Slot Allocation	USN-10	As a User the Customer Care Executive will allocate the slot for me to Scan my car	I can take my car to the Allocated place and time	Medium	Sprint-1
	Estimation	USN-11	As a user now I can know the estimated insurance amount for my car by Scanning	I can know the estimated amount by Scanning	High	Sprint-2
Administrator	Officer	USN-12	As a user know I received the estimated insurance amount by the insurance officer	I can receive the estimated insurance amount	Very High	Sprint-2
	Receipt	USN-13	As a user now I can receive the Receipt for the received insurance amount	I can receive the bill for claiming the insurance	Medium	Sprint-1
Mechanical job	Repairing	USN-14	As a user now I can repair my car by using the insurance amount	I can repair my damaged car	low	Sprint-1

6. PROJECT PLANNING

6.1. Sprint Planning and Estimation

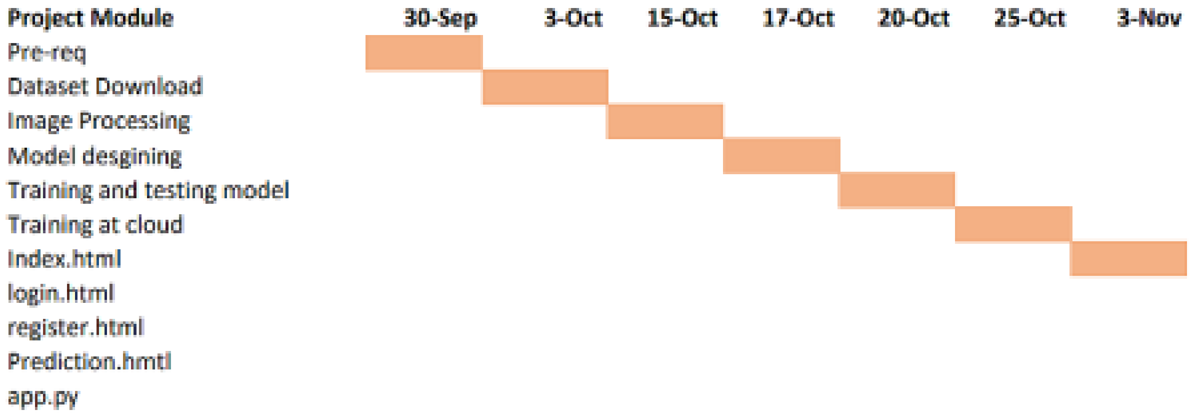


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As an owner of a particular vehicle, I can log into the application by entering email & password.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-1	User Confirmation	USN-2	As an owner of a particular Vehicle, I will receive confirmation email once I have registered for the application.	1	Medium	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-1	Login	USN-3	As an owner of a particular vehicle, I can log into the application by entering email & password.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-2	Data Collection	USN-1	Download the dataset used in intelligent vehicle damage assessment & cost estimator for insurance companies.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R



Sprint-2	Image Pre Processing	USN-1	Improve the image data that suppresses unwilling distortions or enhances some image features important for further processing, although performing some geometric transformations of images like rotation, scaling, etc.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-3	Model Building	USN-1	Define the model architecture and adding CNN layer and testing , saving the model.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-3	Cloud DB	USN-1	Below are steps that need to follow for creating and using cloud ant service. • Register & login to IBM cloud • Create service instance • Creating service credentials • Launch cloud ant DB • Create database	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-4	Application Building	USN-1	Building a web application that is integrated into the model we built. A UI is provided to the user where he has uploaded the image. Based on the saved model, the uploaded image will be analyzed and prediction is showcased on the UI.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R
Sprint-4	Train The Model On IBM	USN-1	Build Deep learning model and computer vision Using the IBM cloud.	2	High	Solai srivarshini R Sri Ranjini N Sruthi V Vidhyasri R

6.2. Sprint Delivery Schedule



Project Module	30-Sep	3-Oct	15-Oct	17-Oct	20-Oct	25-Oct	3-Nov
Pre-req	■						
Dataset Download		■					
Image Processing			■				
Model desgining				■			
Training and testing model					■		
Training at cloud						■	
Index.html							■
login.html							
register.html							
Prediction.hmtl							
app.py							

7. CODING & SOLUTION

7.1. Feature 1

```
client = Cloudant.iam("1c6f917d-87ac-491b-90a0-6e3ae5b5daca
    bluemix","tYJcUyVJYs3WrxF_1absTN4RXrbdQ_RDWBRUy9BX
    28c",connect=True)
database = client.create_database("database_")

#load model
      model1 = load_model('C:\IBM Folder\Vehicle Damage\model\body.h5')
      model2 = load_model('C:\IBM Folder\Vehicle Damage\model\level.h5')
```

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

7.2. Feature 2

```
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]
```

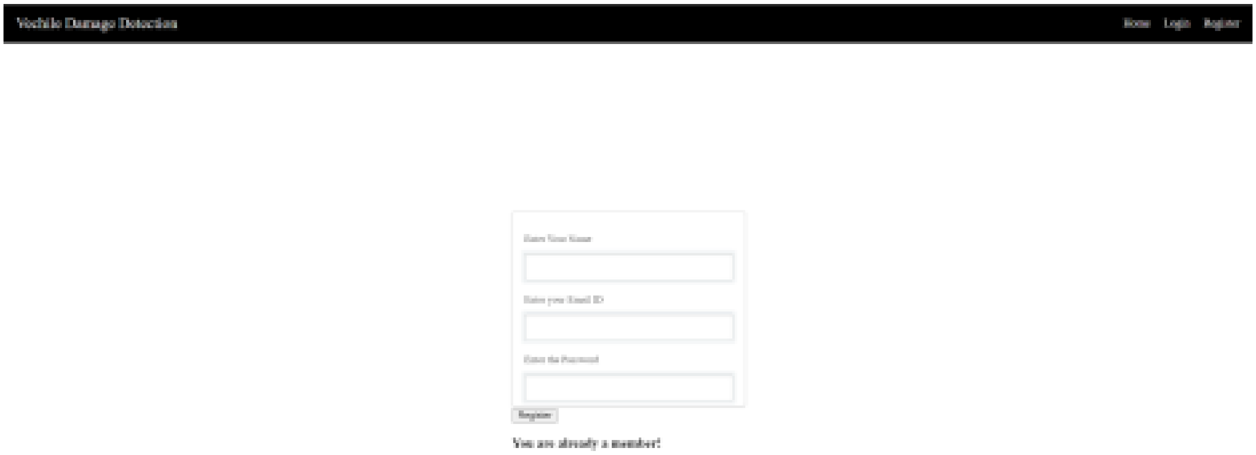
feature 2 enables the web application to predict the incomming image from the user into the given labels. The code gets the image, convert into pixcels and load into the model. Based on the predicted results, the algorithm will returns the value as the estimated cost.

8. TESTING

8.1. Test Cases

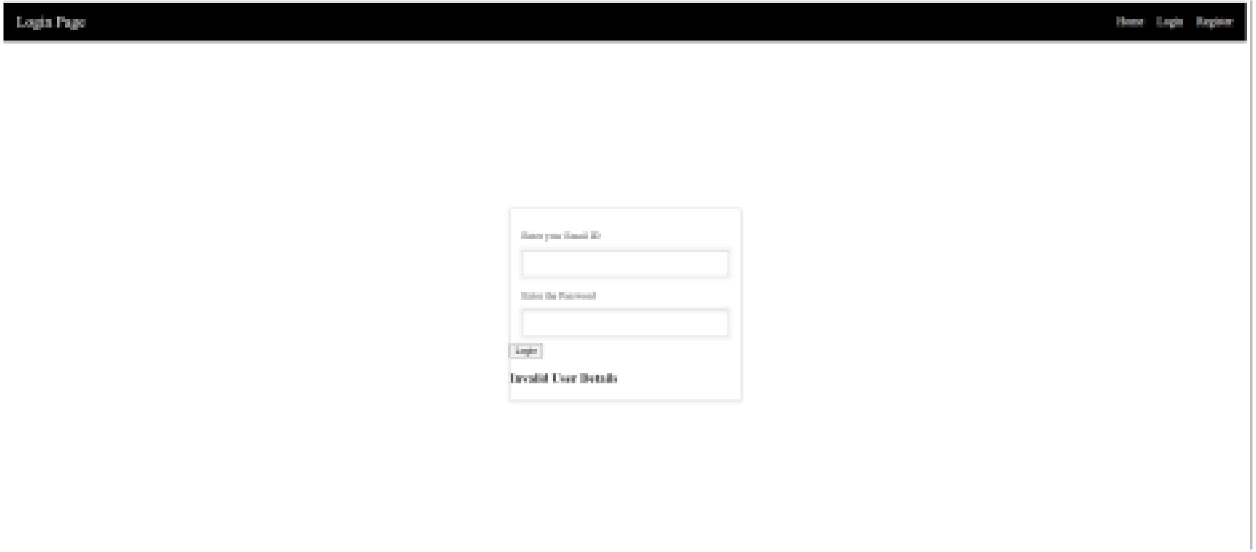
1. User Login and Registration test
2. Database Update test
3. Prediction test

8.2. User Acceptance Testing



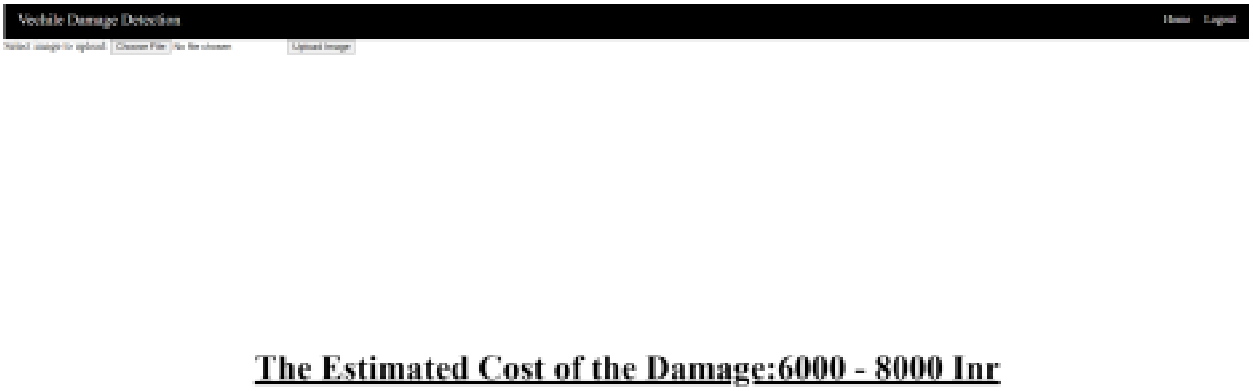
You are already a member!

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



Invalid User Details

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



The Estimated Cost of the Damage:6000 - 8000 Inr

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

9. PERFORMANCE

The performance of the Cost estimator for insurance companies is tested and assested 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 vechile is checked. The results show that the web application took less than 10s to provide the estimated cost of the given vechile 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. ADVANTAGES AND DISADVANTAGES

1. The Advantage of having an Intelligent Cost Estimator based on the damages can save the time and resource of the user in automatically evaluating the images with the damages using the Deep Learning models trained with the various car images.
2. The Disadvantage of the project is expensive coding and time to develop the front end and back end of the web application

11. 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 waston 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.

12. FUTURE SCOPE

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

13. APPENDIX

Github Repo:
https://github.com/IBM-EPBL/IBM-Project-7224-1658850233

VideoLink:
https://drive.google.com/file/d/1wZ_BgP7G8zyOnZ5q5xBCwM7FwBfBapm1/view

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("1c6f917d-87ac-491b-90a0-6e3ae5b5daca
    bluemix","tYJcUyVJYs3WrxF_1absTN4RXrbdQ_RDWBRUy9BX 28c",connect=True)
database = client.create_database("database_")

#load model
model1 = load_model('C:\IBM Folder\Vehicle Damage\model\body.h5')
model2 = load_model('C:\IBM Folder\Vehicle Damage\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

@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"): 15
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)
```