PROJECT REPORT

Emerging Methods for Early Detection of forest fires

Submitted by

PNT2022TMID33183

922119106017 - Chandralekha K

922119106012 - Asleena A

922119106013 - Aswini T

922119106014 - Ayesha Hajera S

TABLE OF CONTENTS

1 INTRODUCTION

1.1 PROJECT OVERVIEW

1.2 PURPOSE

2 LITERATURE SURVEY

2.1 EXISTING PROBLEM

2.2 REFERENCES

2.3 PROBLEM STATEMENT DEFINITION

3 IDEATION AND PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

3.2 IDEATION & BRAINSTORMING

3.3 PROPOSED SOLUTION

3.4 PROBLEM SOLUTION FIT

4 REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

4.2 NON FUNCTIONAL REQUIREMENTS

5 PROJECT DESIGN

5.1 DATA FLOW DIAGRAM

5.2 SOLUTION & TECHNICAL ARCHITECTURE

5.3 USER STORIES

6 PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

6.2 SPRINT DELIVERY SCHEDULE

6.3 REPORTS FROM JIRA

7 CODING & SOLUTIONING

8 TESTING

8.1 TEST CASES

8.2 USER ACCEPTANCE TESTING

8.2.1 DEFECT ANALYSIS

8.2.2 TEST CASE ANALYSIS

9 RESULTS

9.1 PERFORMANCE METRICS

10 ADVANTAGES & DISADVANTAGES

ADVANTAGES

DISADVANTAGES

11 CONCLUSION

12 FUTURE SCOPE

APPENDIX

SOURCE CODE

GITHUB

PROJECT DEMO

CHAPTER 1

INTRODUCTION

1.1Project overview

Forest fires are occurring throughout the year with an increasing intensity in the summer and autumn periods. Forest fires are also considered as a main contributor to the air pollution, due to the fact that during every fire huge amounts of gases and particle mater are released in the atmosphere. In the last decade many improvements in the forest fire detection technologies have been made. The modern IR cameras provide steady and reliable detection of the fires, but the real focus is set on the possibilities to detect the fires by analysing wider areas for smoke or by sensing environmental parameters before the actual spread of the fireTo fight forest fires, different solutions were employed throughout the years. They ware primary aimed at the early detection of the fires.

1.2 Purpose

It is difficult to predict and detect Forest Fire in a sforest area and it is more difficult if the prediction is done using ground-based methods. We have implemented the Artificial Intelligence technology to achieve our objective.

Chapter 2
Literature survey

2.1 Existing Problem

Forest fires are mainly caused by the actions of humans, but different nature and environmental phenomena, like lightning strikes or spontaneous combustion of dried leafs or sawdust, can also be credited for their occurrence. Regardless of the reason for the ignition of the forest fires, they usually cause devastating damage to both nature and humans.

2.2 References

1)Artificial intelligence for forest fire prediction

Authors: George E.Sakr , H.Elhaji , George Mitri , Uchechukwu C.Wejinya

Publication year: July 2010

The proposed method introduces a fire risk index on a scale of 1 to 4, where 1 corresponds to the lowest fire risk and 4 to the highest fire risk. This index is based on the number of fires that occurred on a specific day and hence can be used to estimate the actual number of fires that could happen on that day. The above architecture could be used for monthly prediction, by associating the average weather parameters of a month with a scale for the following month. Or it could be used for annual prediction by associating the average weather of the year with the scale of the next year

2) Early forest fire detection using drones and artificial intelligence

Authors: Diyana Kinaneva, Georgi Hristov, Jordan Raychev, Plamen Zahariev

Publication year: may 2019

If more images are used for the input dataset, the model could become more accurate, but in that case, there is a tradeoff between the model speed and the model accuracy that one must consider. The described system uses machine learning algorithm to monitor the activities of the forest.

3)Title: Case study of forest fire detection systems

Authors: Igorce Karafilovski, Vladimir Zdraveski, Dimitar Trajanov

Publication year: April 2014

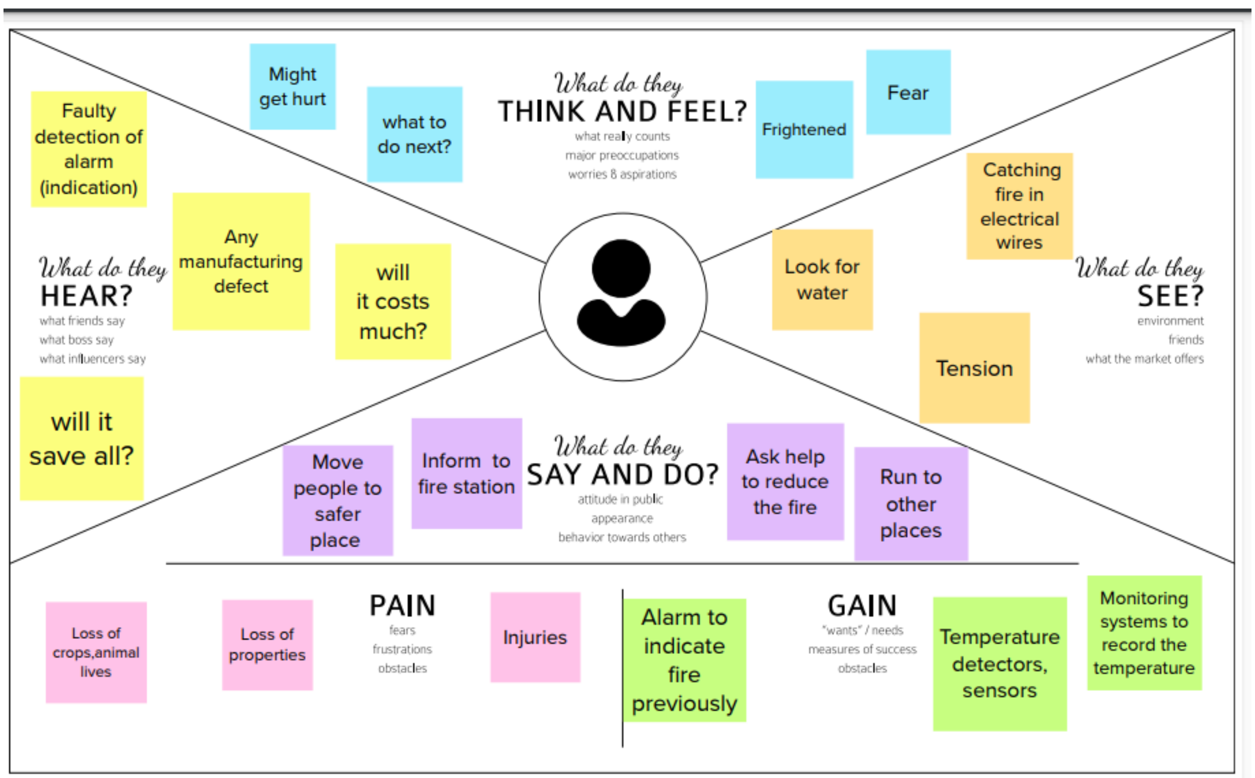
A different approach for a fire forest detection is the integrated system for a forest fire detection. The most effective way to minimize the damages caused by the forest fires is the early detection of forest fires and a fast appropriate reaction. In that direction, in the future, more effective forest fire detection systems need to be developed, that will also utilize the new technologies as smart phones. The appearance of smart phones is a good way to use them as a mobile measuring stations and video detection device.

2.3 PROBLEM SOLUTION STATEMENT

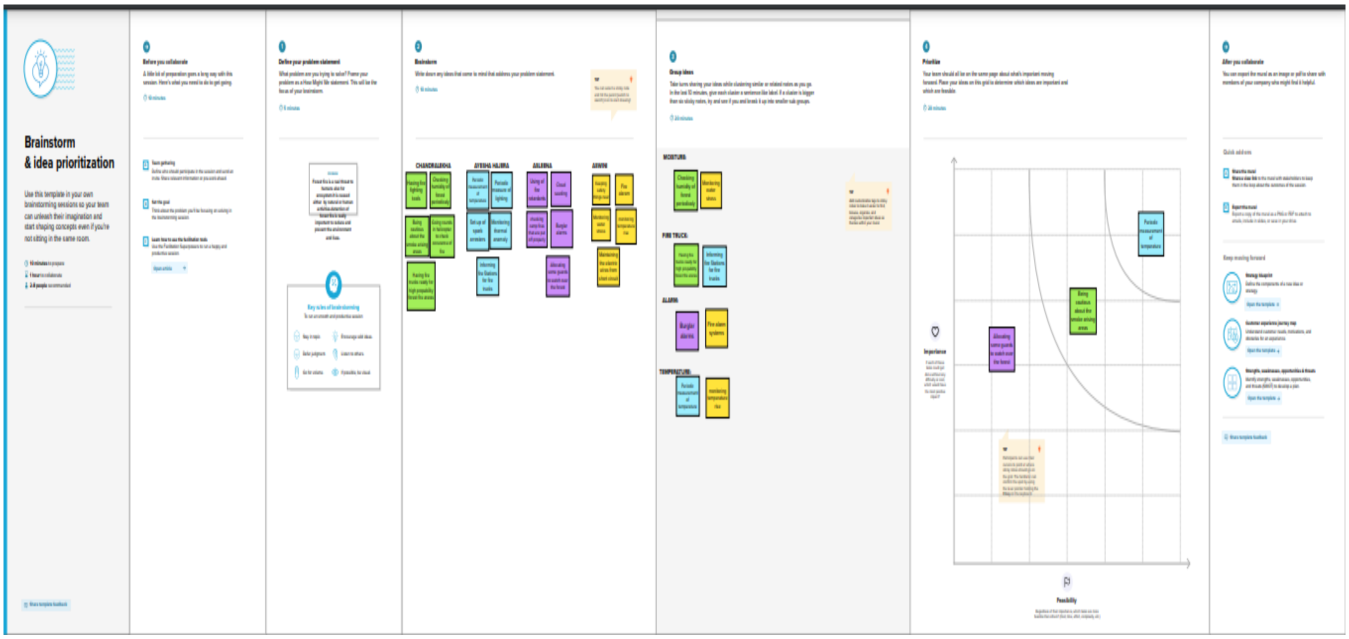
In the current time forest fire is very common which is very destructive and dangerous.It is caused due to the lightning , Temperature rise and other reasons.The forest should be monitored frequently to avoid the fire.It must always have fire fighting tools ready.Alarms should be kept and checked frequently for indication of any fire in the forest.

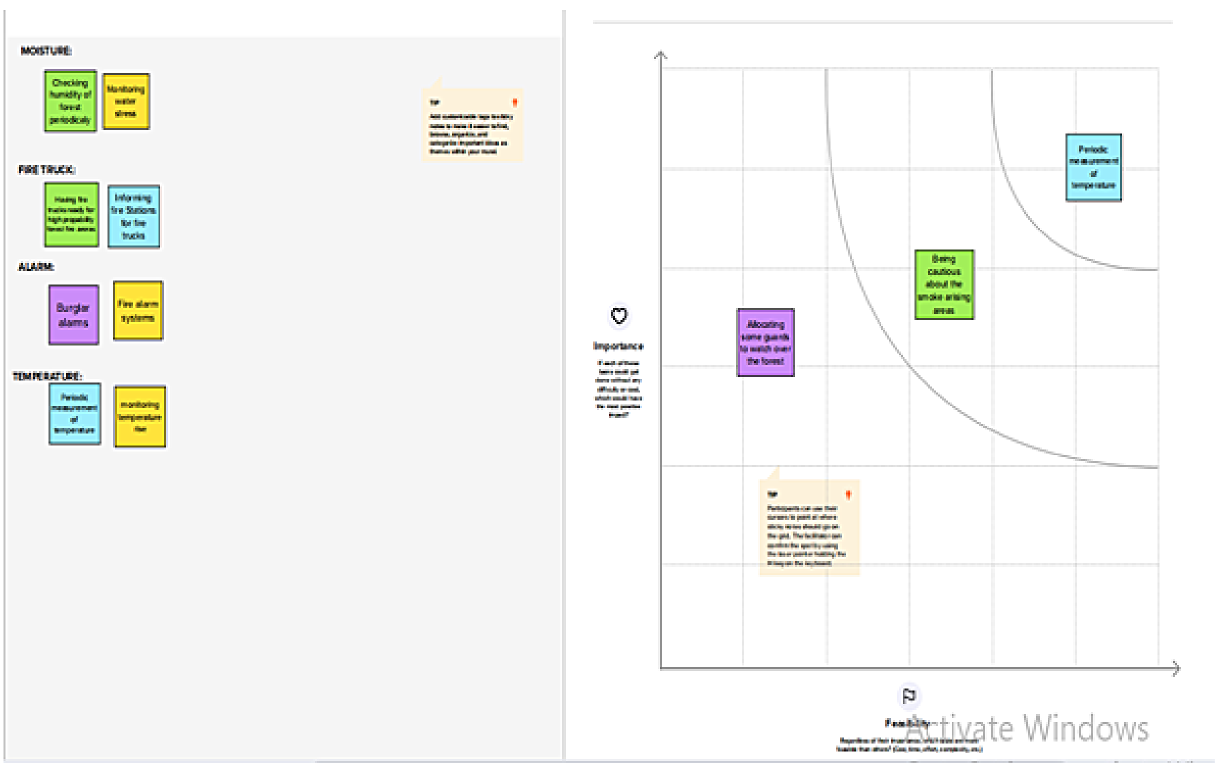
CHAPTER 3
IDEATION AND PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming





3.3 PROPOSED SOLUTION

S.No	Parameter	Description
1	Problem Statement (Problem to be solved)	• Forest fires are an integral part of many terrestrial ecosystems. Every year, thousands of forest fire across the globe cause disasters beyond measure and description. • Forest fire prediction, prevention and management measures have become increasingly important.
2	Idea / Solution description	• The computer vision methods for recognition and detection of smoke and fire, based on the still images or the video input from the cameras. Deep learning method “convolution neural network” can be used for finding the amount of fire. • Enabling the video surveillance systems on forest to handle more complex situations in real world. • Accuracy for detection of fire can be given based on the algorithm which we are going to use and the datasets and splitting them into train set and test set.

3	Novelty / Uniqueness	• Using the algorithm like CNN, the pre-processing required in a ConvNet is much lower as compared to other classification algorithms. • Parameters such as temperature, gas concentrations, soil humidity etc. are monitored with sensors while background sounds are analysed
4	Social Impact / Customer Satisfaction	• The customer will not be informed with a false alarm. To avoid the instances of false alarms being triggered, a threshold for the classifier confidence was set. Hence, the alarm is only triggered when the confidence is greater or equal to the threshold. • The flame detection frameworks can be intelligently tuned for detection of fire. This will enable the video surveillance systems on forest to handle more complex situations in real world.

5	Business Model (Revenue Model)	• Finally built fire detection unit becomes inevitable to use a test dataset that includes images that are often encountered in real-world fire emergencies with an image quality that is commonly obtained with a camera attached to low-cost hardware like Arduino Uno, a microcontroller. • It will be very useful for the customers who are buying as well as the manufacturers.
6	Scalability of the Solution	• Many fire risk models make use of forest fire databases to construct and assess probabilistic models. • It is an effective way to minimize the damages caused by the forest fires is the early detection of forest fires and a fast appropriate reaction. • As a scalable model, in the future, it is vital to study the temporal and spatial distributions of forest microclimate related factors indepth, analyse the law of coordinated changes and combine microclimate factors in the model simulation and training processes.

3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Forest guard

6. CUSTOMER CONSTRAINTS

Spending more money for the equipments, network connection for the devices, power supply interruptions, occurrence of damages sometimes these limitations the customers choices of solutions .

5. AVAILABLE SOLUTIONS

Alarm system for indication of five, remote sensing based methods such as satellites, high -resolution static cameras fixed on the ground, unmanned aerial vehicles.

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

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

Always clear the area around the workspace.

The area should be even larger if it is windy and dry.

Making sure that to never operate equipment that produces sparks near dry vegetation.

9. PROBLEM ROOT CAUSE RC

The fire is mainly caused by lightning, increased temperature, human activities and other reasons .

Human caused fires result from campfires, equipment use and malfunction, negligently discarded cigarettes, etc..

7. BEHAVIOUR BE

They to monitor the forest areas themselves, often checking whether the camp fire are put off properly.

Always having fire fighting tools always ready.

Monitoring the temperature in the forest.

Focus on J&P, tap into BE, understand RC

3. TRIGGERS TR

The need to protect the wildlife and themselves triggers them to act.

Not knowing when would fire starts

Taking suggestion from visitors.

4. EMOTIONS: BEFORE / AFTER EM

They don't feel safe.

Always fear of catching fire in the forest.

Panic at the of sudden forest fire.

Afterwards:

They will have some satisfaction of knowing that some indication will come on the start of fire.

10. YOUR SOLUTION SL

The computer vision methods for recognition and detection of smoke and fire, based on the still images or the video input from the cameras.

Deep learning method "convolution neural network" can be used for finding the amount of fire.

Enabling the video surveillance systems on forest to handle more complex situations in real world.

8.CHANNELS of BEHAVIOUR CH

Online:

Installing cameras and sensors in parts of the forest and checking the situation.

Offline:

Making sure that no fire is started near the dry plants or highly inflammable objects.

CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional requirements

Following are the functional requirements of the proposed solution.

S.No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	Dataset	Splitting the dataset into training set and testing set .So that it can be used for detecting the occurrence of fire if any.
FR-2	Continuous monitoring system	It is useful for monitoring. Whether there is occurrence of fire or not. Continuous monitoring system helps in this way. This protects the people and other animals in the forest and near the forest from getting damage.
FR-3	IBM cloud	Then the Convolution Neural Network (CNN) model is deployed on the IBM. By using the convolution neural network the images taken by the monitoring system will be able to detect the fire occurrence.
FR-4	Twilio service	Installing the twilio service. Whenever the fire is detected in the forest with the help of the cnn model, it will help to send messages to alert the clients about the fire in the forest.

Non-functional Requirements:

Following are the non-functional requirements of the proposed solution

FR No	Non-Functional Requirement	Description
NFR-1	**Usability**	It is very easily usable for the customer. They get a message whenever the forest fire is detected. It makes them easier to monitor the activities in the forest areas as well as also in the areas of high fire risking. Use of convolution neural network makes it

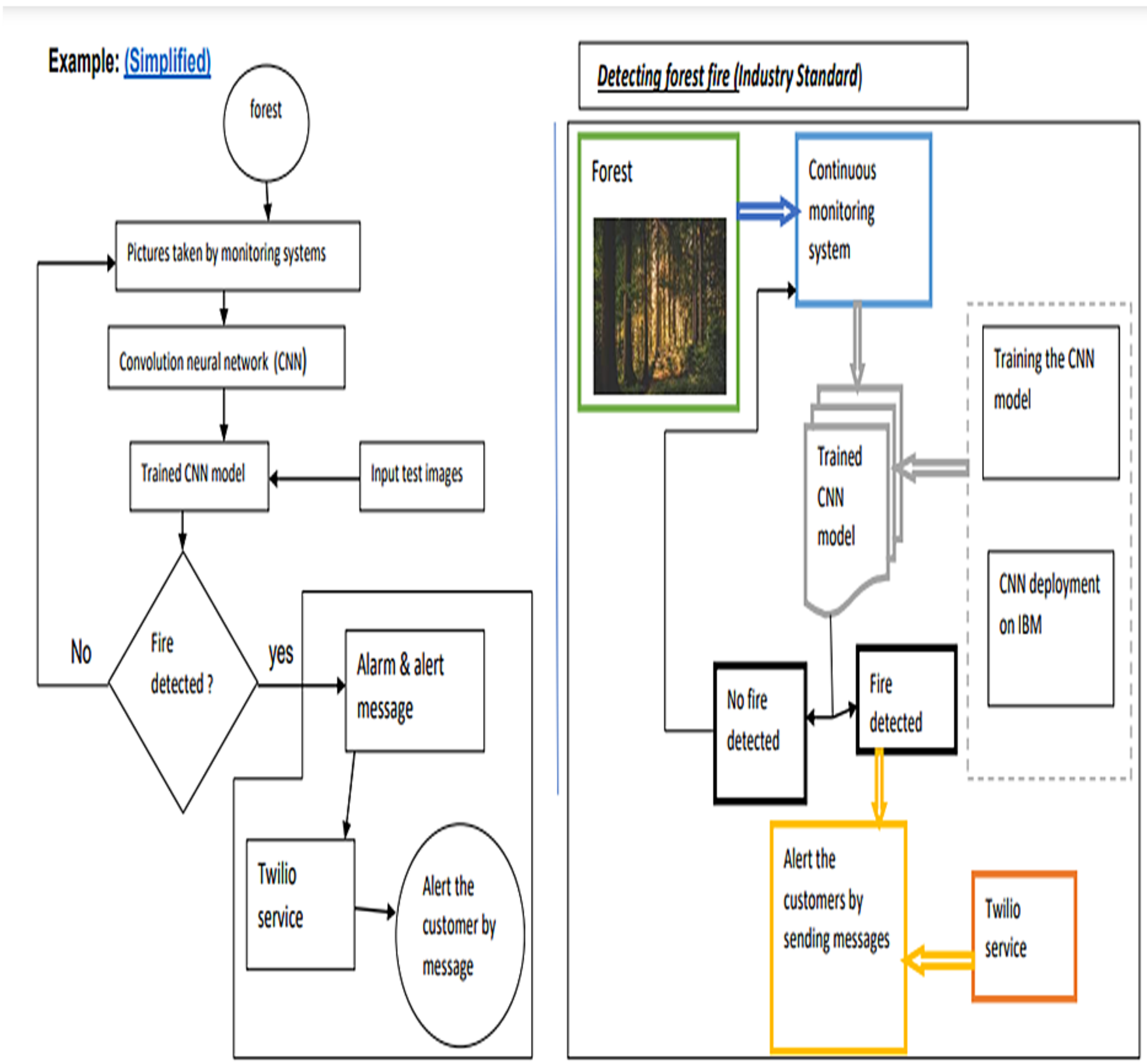
NFR-2	**Security**	Security is very much concerned regarding the datas collected and customer details . These securities are mainly related to the cloud services, they have strict security across the network. And the twilio service also maintains standards to ensure data is appropriately stored, as security is the most important matter in all the fields of technology now a days
NFR-3	**Reliability**	This method of using the artificial intelligence gives appropriate results. And the security makes sure that the details and data collected are safe. The relaiability is more for the customers.

NFR-4	**Performance**	After repeated training and testing ,the forest fire prediction results based on the convolution neural network would be found to be appropriate in most of the times. When the fire in the forest is detected the alerting message will be sent to the customers(clients). So that they can prevent the major damages caused by the fire.
NFR-5	**Availability**	As the mentioned functional requirements are mostly open sources, they are highly available to all. Anyone can make use of it.
NFR-6	**Scalability**	Many fire risk models make use of forest fire databases to construct and assess the probabilistic model. It is an effective way to minimize the damages caused by forest fire in the early detection of forest fire.

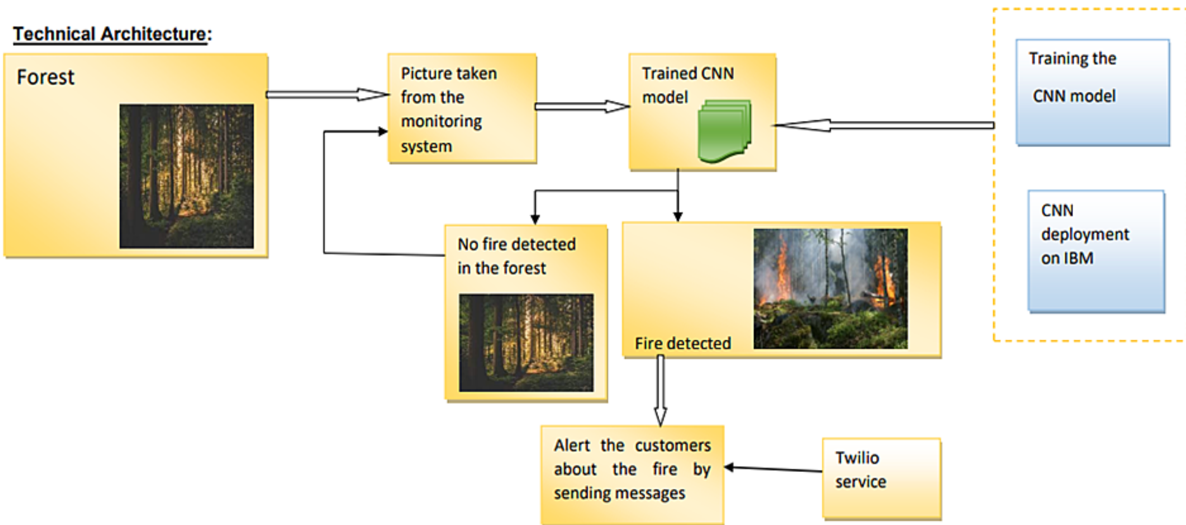
CHAPTER 5

PROJECT DESIGN

5.1 Data Flow Diagram



5.2 SOLUTION AND TECHNICAL ARCHITECTURE



5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Developer	Registration	USN-1	As a user, I can sign up and register respective sites to access the required details and data. And import the required libraries for the processes.	I can access the account / dashboard	High	Sprint-1
Assistant developer	Login	USN-2	As a user, I will access the page and test and train the CNN model to predict or detect the forest fire.	I can test and confirm the error free detections	High	Sprint-2

Customer Care Executive	Worker	USN-3	As a customer care executive ,i am available to the customers .so if the customers have any issues or in need of any assistance they will get help and solve them	I can be in contact with the customers.	medium	Sprint-3
Customer (Web user)	Login	USN-4	As a user , i will have the access to know about the activities in the forest.	I can get messages when there is fire in the forest	High	Sprint-4

CHAPTER 6

PROJECT PLANNING AND SCHEDULING

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 a user, I can sign up and register respective sites to access the required details and data. And import the required libraries for the processes.	2	High	Asleena Ayesha hajera
Sprint-2	Login	USN-2	As a user,I will access the page and test and train the CNN model to predict or detect the forest fire.	2	High	Chandralekha, Aswini .

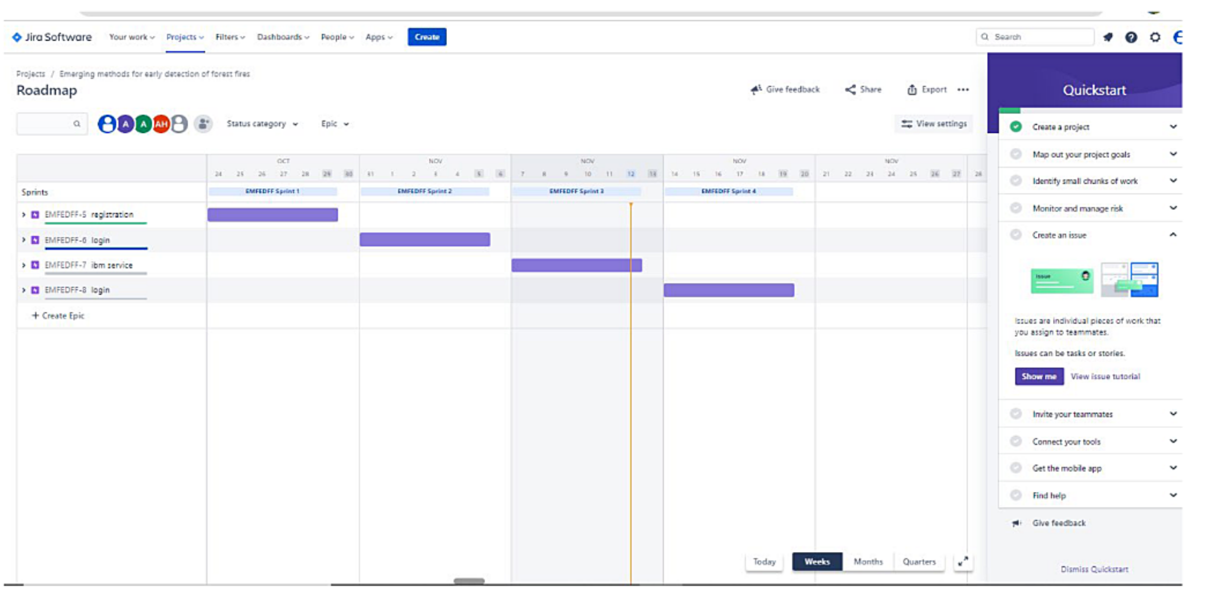
Sprint-3	Ibm service	USN-3	As a user, I can access the ibm cloud and deploy the trained CNN model. So the deployed model can be accessed whenever wanted.	1	Medium	Asleena , Chandralekha
Sprint-4	Login	USN-4	As a user, I can analysis the activites via video analysis whether there is fire occurrence or not.	2	High	Aswini , Ayesha hajera

6.2 SPRINT DELIVERY SCHEDULE

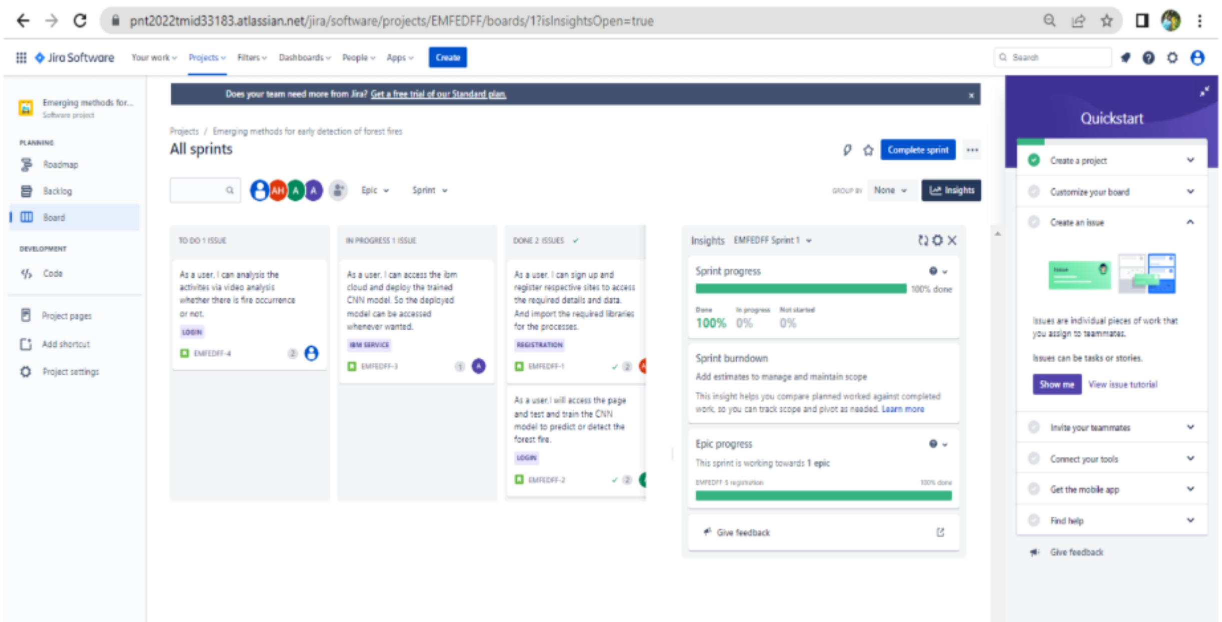
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	4th Nov 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	6th Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	20	9th Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	12th Nov 2022

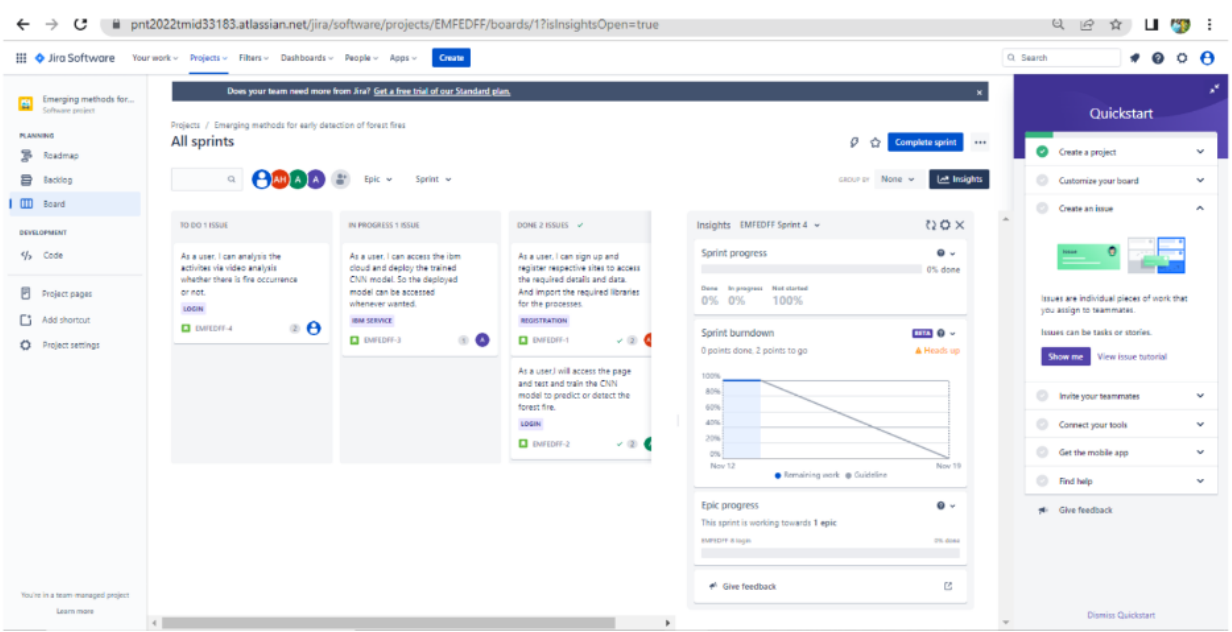
6.3 REPORTS FROM JIRA

Using JIRA software

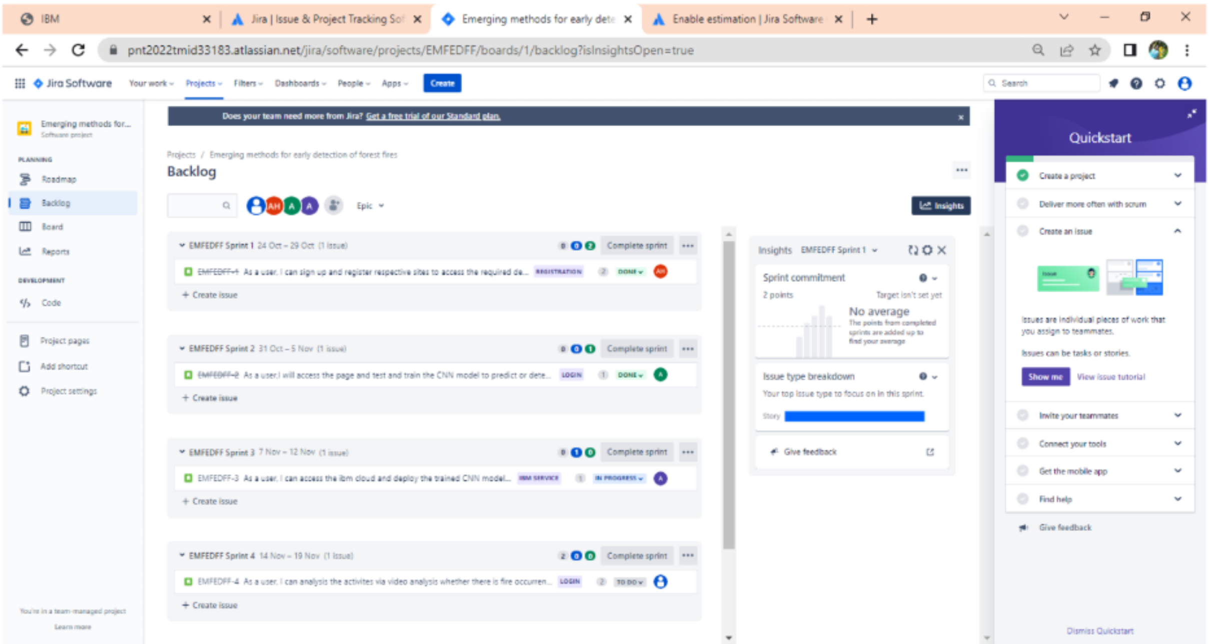


Burndown chart:





VELOCITY CHART



CHAPTER 7

CODING & SOLUTIONING

Model building & Saving

```
#importing the necessory libraries

from tensorflow.keras.preprocessing.image import ImageDataGenerator

#define the parameters/arguments for imagedatagenarator class
train_datagen=ImageDataGenerator(rescale=1./255,
                                  shear_range=0.2,
                                  rotation_range=180,
                                  zoom_range=0.2,
                                  horizontal_flip=True)


test_datagen=ImageDataGenerator(rescale=1./255)

#applying imagedatagenerator functionality to trainset
x_train=train_datagen.flow_from_directory('/content/Dataset/Dataset/
train_set',
                                          target_size=(128,128),
                                          batch_size=32,
                                          class_mode='binary')

Found 436 images belonging to 2 classes.

#applying imagedatagenerator functionality to testset
x_test=test_datagen.flow_from_directory('/content/Dataset/Dataset/
test_set',
                                        target_size=(128,128),
                                        batch_size=32,
                                        class_mode='binary')


Found 121 images belonging to 2 classes.

#Importing the model libraries

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D, MaxPooling2D,
Flatten, Dense

#initializing the model
model=Sequential()

#adding CNN layers
model.add(Convolution2D(32,
(3,3),input_shape=(128,128,3),activation='relu'))    # convolution
layer
```

```
#adding CNN layers
model.add(Convolution2D(32,
(3,3),input_shape=(128,128,3),activation='relu'))    # convolution
layer
model.add(MaxPooling2D(pool_size=(2,2)))
#maxpool layer
model.add(Flatten())
#flatten layer

#adding dense layers
model.add(Dense(150,activation='relu'))
#hidden layer
model.add(Dense(150,activation='relu'))
#hidden layer

model.add(Dense(1,activation="sigmoid"))
#output layer


#configuring the learning process
model.compile(loss='binary_crossentropy',optimizer='adam',
              metrics=['accuracy'])
```

```python
#training the model
model.fit_generator(x_train,
                    steps_per_epoch=14,
                    epochs=10,
                    validation_data=x_test,
                    validation_steps=4)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:6:
UserWarning: `Model.fit_generator` is deprecated and will be removed
in a future version. Please use `Model.fit`, which supports
generators.
```

```
Epoch 1/10
14/14 [==============================] - 27s 2s/step - loss: 1.7052 -
accuracy: 0.6055 - val_loss: 0.3188 - val_accuracy: 0.8512
Epoch 2/10
14/14 [==============================] - 25s 2s/step - loss: 0.3500 -
accuracy: 0.8417 - val_loss: 0.1166 - val_accuracy: 0.9504
Epoch 3/10
14/14 [==============================] - 25s 2s/step - loss: 0.1948 -
accuracy: 0.9106 - val_loss: 0.1255 - val_accuracy: 0.9669
Epoch 4/10
14/14 [==============================] - 25s 2s/step - loss: 0.1806 -
accuracy: 0.9106 - val_loss: 0.0860 - val_accuracy: 0.9752
Epoch 5/10
14/14 [==============================] - 25s 2s/step - loss: 0.1612 -
accuracy: 0.9289 - val_loss: 0.1147 - val_accuracy: 0.9669
Epoch 6/10
14/14 [==============================] - 25s 2s/step - loss: 0.1650 -
accuracy: 0.9381 - val_loss: 0.0723 - val_accuracy: 0.9752
Epoch 7/10
14/14 [==============================] - 25s 2s/step - loss: 0.1489 -
accuracy: 0.9472 - val_loss: 0.0692 - val_accuracy: 0.9504
Epoch 8/10
14/14 [==============================] - 25s 2s/step - loss: 0.1458 -
accuracy: 0.9381 - val_loss: 0.0671 - val_accuracy: 0.9835
Epoch 9/10
14/14 [==============================] - 25s 2s/step - loss: 0.2015 -
accuracy: 0.9106 - val_loss: 0.0563 - val_accuracy: 0.9752
Epoch 10/10
14/14 [==============================] - 25s 2s/step - loss: 0.1710 -
accuracy: 0.9266 - val_loss: 0.0585 - val_accuracy: 0.9835

<keras.callbacks.History at 0x7fc0f7f46550>
```

```python
#saving the model
model.save("forest2.h5")
```

```
#predicting
from keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np

img=image.load_img('/content/forest fire
(1).jpg',target_size=(128,128))          #prediction-1
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
result=model.predict(x)
if result[0][0]==1:
  prediction='forest with fire'
  print(prediction)
else:
  prediction='forest without fire'
  print(prediction)
```

```
1/1 [==============================] - 0s 36ms/step
forest with fire
```

```
img=image.load_img('/content/Dataset/Dataset/test_set/forest/
091318_LH_forest_loss_main_FREE.jpg',target_size=(128,128))
#prediction-2
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
result=model.predict(x)
if result[0][0]==1:
  prediction='forest with fire'
  print(prediction)
else:
  prediction='forest without fire'
  print(prediction)
```

```
1/1 [==============================] - 0s 37ms/step
forest without fire
```

```
img=image.load_img('/content/Dataset/Dataset/test_set/with
fire/How_to_Protect_Your_Home_From_Forest_Fire_1024x588.jpg',target_si
ze=(128,128))  #prediction-3
x = image.img_to_array(img)
x = np.expand_dims(x,axis=0)
result=model.predict(x)
if result[0][0]==1:
  prediction='forest with fire'
  print(prediction)
else:
  prediction='forest without fire'
  print(prediction)
```

```
1/1 [==============================] - 0s 50ms/step
forest with fire
```

x_train.class_indices

{'forest': 0, 'with fire': 1}

Video Analysis

video analysis using opencv

#AFTER ANALYSING THE VIDEO THE ALERT MESSAGE IS SENT TO THE CORRESPONDING NUMBER WHEN THE FIRE IS DETECTED

after training and saving the model ..here we used the saved model directly .

the model is loaded and then tested with video having fire and video having no fire

```
import cv2
import numpy as np
from google.colab.patches import cv2_imshow
from matplotlib import pyplot as plt
import librosa
from tensorflow.keras.preprocessing import image
from keras.models import load_model
from twilio.rest import Client
from playsound import playsound
```

```
WARNING:playsound:playsound is relying on another python subprocess.
Please use `pip install pygobject` if you want playsound to run more
efficiently.
```

```
cap = cv2.VideoCapture('/content/forest with fire.mp4')
if (cap.isOpened()==False):
  print('video streaming or file can not be opened.error occured ')
while(cap.isOpened()):
  ret,frame=cap.read()
  if ret ==True:
    x=image.img_to_array(frame)
    res=cv2.resize(x,dsize=(64,64),interpolation=cv2.INTER_CUBIC)
```

```
    x=np.expand_dims(res,axis=0)
    model=load_model('/content/forest1.h5')
    cv2_imshow(frame)
    pred=model.predict(x)
    pred=int(pred[0][0])
    pred
    int(pred)
    if pred==0:
      print('no forest fire')
      break
    else:
      print('forest fire')
      break
cap.release()
cv2.destroyAllWindows()
```

```
1/1 [==============================] - 0s 68ms/step
forest fire
```

```
cap = cv2.VideoCapture('/content/forest without fire.mp4')
if (cap.isOpened()==False):
  print('video streaming or file can not be opened.error occured ')
while(cap.isOpened()):
```

```
  ret,frame=cap.read()
  if ret ==True:
    x=image.img_to_array(frame)
    res=cv2.resize(x,dsize=(64,64),interpolation=cv2.INTER_CUBIC)
    x=np.expand_dims(res,axis=0)
    model=load_model('/content/forest1.h5')
    cv2_imshow(frame)
    pred=model.predict(x)
    pred=int(pred[0][0])
    pred
    int(pred)
    if pred==0:
      print('no forest fire')
      break
    else:
      print('forest fire')
      break
cap.release()
cv2.destroyAllWindows()
```

```
WARNING:tensorflow:5 out of the last 5 calls to <function
Model.make_predict_function.<locals>.predict_function at
0x7f66fc5b0b90> triggered tf.function retracing. Tracing is expensive
and the excessive number of tracings could be due to (1) creating
@tf.function repeatedly in a loop, (2) passing tensors with different
shapes, (3) passing Python objects instead of tensors. For (1), please
define your @tf.function outside of the loop. For (2), @tf.function
has reduce_retracing=True option that can avoid unnecessary retracing.
For (3), please refer to
https://www.tensorflow.org/guide/function#controlling_retracing and
https://www.tensorflow.org/api_docs/python/tf/function for  more
details.

1/1 [==============================] - 0s 71ms/step
no forest fire
```

#SENDING ALERT MESSAGES

```
cap = cv2.VideoCapture('/content/forest with fire.mp4')
if (cap.isOpened()==False):
  print('video streaming or file can not be opened.error occured ')
while(cap.isOpened()):
  ret,frame=cap.read()
  if ret ==True:
    x=image.img_to_array(frame)
    res=cv2.resize(x,dsize=(64,64),interpolation=cv2.INTER_CUBIC)
    x=np.expand_dims(res,axis=0)
    model=load_model('/content/forest1.h5')
```

```
  ret,frame=cap.read()
  if ret ==True:
    x=image.img_to_array(frame)
    res=cv2.resize(x,dsize=(64,64),interpolation=cv2.INTER_CUBIC)
    x=np.expand_dims(res,axis=0)
    model=load_model('/content/forest1.h5')
    cv2_imshow(frame)
    pred=model.predict(x)
    pred=int(pred[0][0])
    pred
    int(pred)
    if pred==0:
      print('no forest fire')
      break
    else:
      #sending alert message when fire is detected
      account_sid='AC1bde03c3f8ea537b598ad65f2ce62d5d'
      auth_token='7954ac5c2458f8d96d**************'
      client=Client(account_sid,auth_token)
      message=client.messages \
      .create(
          body='forest fire occurance is detected,stay safe and
alert',
          from_='+1606517****',
          to='+9188381*****'
      )
      print(message.sid)
```

```
        print('forest fire')
        break
cap.release()
cv2.destroyAllWindows()
```

```
1/1 [==============================] - 0s 58ms/step
SM05a0642886e5af91d23a52a779d2fade
```

forest fire

```
cap = cv2.VideoCapture('/content/forest without fire.mp4')
if (cap.isOpened()==False):
  print('video streaming or file can not be opened.error occured ')
while(cap.isOpened()):
  ret,frame=cap.read()
  if ret ==True:
    x=image.img_to_array(frame)
    res=cv2.resize(x,dsize=(64,64),interpolation=cv2.INTER_CUBIC)
    x=np.expand_dims(res,axis=0)
    model=load_model('/content/forest1.h5')
    cv2_imshow(frame)
    pred=model.predict(x)
    pred=int(pred[0][0])
    pred
    int(pred)
    if pred==0:
      print('no forest fire')
      break
    else:
      #sending alert message when fire is detected
      account_sid='AC1bde03c3f8ea537b598ad65f2ce62d5d'
      auth_token='7954ac5c2458f8d96d*************'
      client=Client(account_sid,auth_token)


      message=client.messages \
      .create(
          body='forest fire occurance is detected,stay safe and
alert',
          from_='+1606517****',
          to='+9188381*****'
      )
      print(message.sid)
      print('forest fire')
      break
cap.release()
cv2.destroyAllWindows()
```

```
1/1 [==============================] - 0s 72ms/step
no forest fire
```

CHAPTER 8

TESTING

8.1 TEST CASES

TEST CASES REPORT

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	BUG ID	Executed By
Test_case_001	Functional	Page	Check if user can upload their file	Internet connection, Laptop/mobile	1.Enter URL and click go	File having Forest fire content	The input file should be uploaded successfully	Working as expected	PASS		K.Chandralekha T.Aswini
Test_case_002	Functional	Page	Check if user cannot upload unsupported files	Internet connection, Laptop/mobile	1.Enter URL and click go	File.zip	The application should show "the file can not be opened" for unsupported files	Working as expected	PASS		A.Asleena S. Ayesha Hajera
Test_case_003	Functional	Backend	Checks whether the model can handle various image sizes	Internet connection, Laptop/mobile	1.Enter URL and click go Open the page Upload the input images	File having without fire	The model should rescale the file and predict the results	Working as expected	PASS		k.chandralekha S. Ayesha Hajera
Test_case_004	UI	model	Checks if the model can predict the forest fire occurence	Internet connection, Laptop/mobile	Select the input file and upload	1)file with forest fire 2) file with without forest fire	The prediction should be displayed properly	Working as expected	PASS		T.Aswini A.Asleena
Test_case_005	API	Result Page	Sends alert message when the forest fire is detected	Internet connection, Laptop/mobile	Select the input file and upload	File having forest fire	Message received	Working as expected	PASS		k.chandralekha T.Aswini

8.2 USER ACCEPTANCE TESTING

1. PURPOSE OF DOCUMENT

The purpose of this document is to briefly explain the test coverage and open issues of the emerging methods for early detection of forest fires project at the time of the release to User Acceptance Testing (UAT).

2.DEFECT ANALYSIS

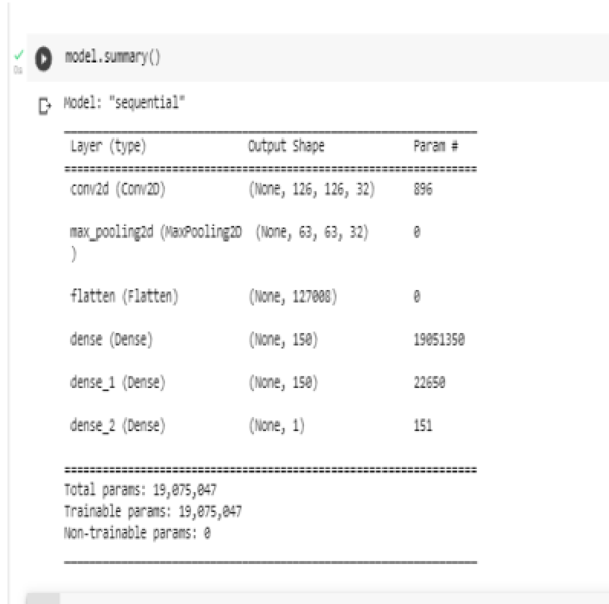
Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	1	1	1	0	3
Duplicate	0	0	0	0	0
External	0	0	2	0	2
Fixed	3	1	0	1	5
Not Reproduced	0	0	0	1	1
Skipped	1	0	1	0	2
Won't Fix	1	0	0	0	1
Totals	6	2	4	2	14

8.3 TEST CASE ANALYSIS

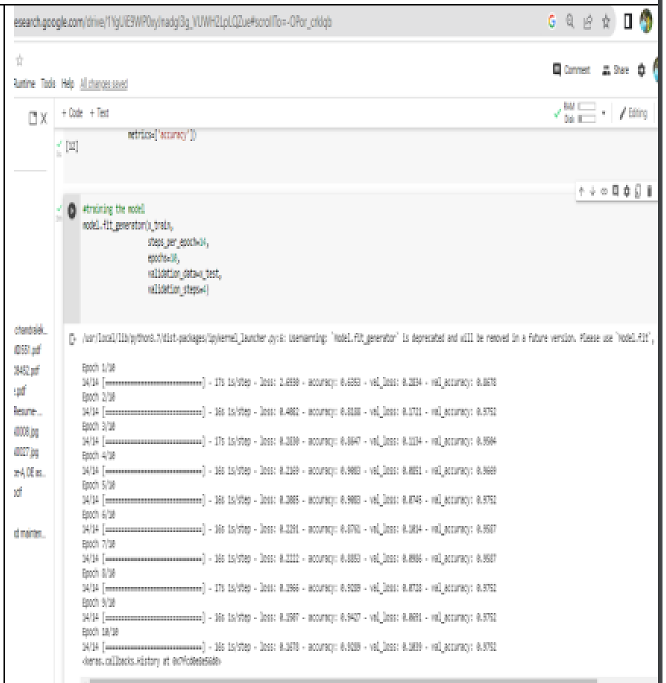
Section	Total Cases	Not Tested	Fail	Pass
Client Application	5	0	0	5
Security	2	0	0	2
Performance	2	0	0	2
Exception Reporting	1	0	0	1

CHAPTER 9
RESULTS

9.1 PERFORMANCE METRICS

No.	Parameter	Values	Screenshot
1.	Model Summary	Total params:19, 075,047 Trainable params:19,075,047 Non-trainable params:0	

```
model.summary()

Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 126, 126, 32)      896

 max_pooling2d (MaxPooling2D  (None, 63, 63, 32)       0
 )

 flatten (Flatten)           (None, 127008)            0

 dense (Dense)               (None, 150)               19051350

 dense_1 (Dense)             (None, 150)               22650

 dense_2 (Dense)             (None, 1)                 151

=================================================================
Total params: 19,075,047
Trainable params: 19,075,047
Non-trainable params: 0
_________________________________________________________________
```

2.	Accuracy	Training Accuracy – 0.9289 Validation Accuracy -0.9752	
3.	Confidence Score (Only Yolo Projects)	Class Detected - nill Confidence Score -nill	No YOLO

CHAPTER 10

ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Today, our everyday lives are entirely dependent on mobile devices and the internet. Today, our everyday lives are entirely dependent on mobile devices and the internet.
- With the use of various AI-based techniques, we can also anticipate today's weather and the days ahead.
- Artificial Intelligence detects the fire easily by using the temperature sensors.
- The alert message is received early through our project to the user.

DISADVANTAGES

- AI is capable of learning over time with pre-fed data and past experiences, but cannot be creative in its approach.
- Manufacturing defect may cause high impact to human beings.
- Ethics and morality are important human features that can be difficult to incorporate into an AI. The rapid progress of AI has raised a number of concerns that one day, AI will grow uncontrollably, and eventually wipe out humanity. This moment is referred to as the AI singularity.

CHAPTER 11

CONCLUSION

In this project we have briefly presented methods for early forest fire detection, including part of their characteristics and main components. We have also analysed some of the benefits. Monitoring of the potential risk is an early detection of fire can significantly shorten the reaction time and also reduce the potential damage as well as the cost of fire fighting.

CHAPTER 12

FUTURE SCOPE

AI in Cyber Security

Cybersecurity is another field that’s benefitting from AI. As organizations are transferring their data to IT networks and cloud, the threat of hackers is becoming more significant.

AI in Data Analysis

Data analysis can benefit largely from AI and ML. AI algorithms are capable of improving with iterations, and this way, their accuracy, and precision increase accordingly. AI can help data analysts with handling and processing large datasets.

AI in Home

AI has found a special place in people’s homes in the form of Smart Home Assistants. Amazon Echo and Google Home are popular smart home devices that let you perform various tasks with just voice commands.

AI in Healthcare

The medical sector is also using this technology for its advantages. AI is helping medical researchers and professionals in numerous ways.

AI in Education

The importance of education in this world has been prevalent, but it continues to grow even today. With a large part of the country's population being the youth, it is important that they receive a good quality education. Along with that, it is also necessary that they understand AI and its benefits. Just like all the other sectors, it is critical for the education sector to keep up with AI as well as the artificial intelligence scope keeps increasing to fuel the education sector.

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-2222-1658466999

DEMO LINK

https://youtu.be/XnykqN3pVjI