# Emerging Methods for Early Detection ofForest Fires

## PROJECT REPORT

***Submitted by***

## *TEAM ID:* PNT2022TMID36539

**TEAM MEMBERS**

**TEAM LEAD : N.S. NITHIKSHA(112619104011)**

**TEAM MEMBER 1: V. HARINI(112619104005)**

**TEAM MEMBER 2: S. THEEBAN BABU(112619104014)**

**TEAM MEMBER 3: M. VIKRAM(112619104016)**

# TABLE OF CONTENTS

# TABLE OF CONTENTS

# 1 INTRODUCTION

## 1.1 PROJECT OVERVIEW

Forest fires are a major environmental issue, creating economic and ecological damage while endangering human lives. There are typically about 100,000 wildfires in the United States every year. Over 9 million acres of land have been destroyed due to treacherous wildfires. Regardless of the reasons for the ignition of the forest fires, they normally cause withering damage to both nature and humans.

Forest fires are also considered as a main contributor to the air pollution, due to the fact that during every fire huge volume of gases and particle mater are released in the atmosphere. To fight forest fires, different solutions were employed throughout the years. They ware primary aimed at the early detection of the fires. The simplest of these solutions is the formation of a network of observance posts - both cheap and easy to accomplish, but also time-consuming for the involved people.

It is difficult to predict and detect Forest Fire in a sparsely populated forest area and it is more difficult if the prediction is done using ground-based methods like Camera or Video-Based approach. Satellites can be an important source of data prior to and also during the Fire due to its reliability and efficiency. The various real-time forest fire detection and prediction approaches, with the goal of informing the local fire authorities.

The constant evolution of the info and communication applied science has led to the debut of a new generation of solutions for early detection and even prevention of forest fires. ICT-based networks of cameras and sensors and even satellite-based solutions were developedand used in the last decades. These solutions have greatly decreased the direct involvement of humans in the forest fire detection process, but have also proven to be expensive and hard to maintain.

## 1.2 PURPOSE

Detection of forest fire and smoke in wild land areas is done through remote sensing- based methods such as satellites, high-resolution static cameras fixed on the ground, and unmanned aerial vehicles Optical/thermal cameras deployed on the observation towers togetherwith the other sensors such as smoke, temperature, and humidity sensors might detect the hazards in the closed environment rather than in the open environment as these sensors need vicinity to the fire or smoke.

The information obtained through these sensors is not appropriate. Distance covered by these methods could be limited, and to cover a large area, more sensors have to be deployed thatmight incur expenses. Through the deployment of UAV, large areas could be covered, and the images with high spatial and temporal resolutions could be captured properly.

# 2 LITERATURE SURVEY

## 2.1 EXISTING PROBLEM

### Literature survey-1

Different fire detection techniques have been proposed for safety and protection of the people and environment.

- Wavelet based smoke detection-smoke detection in video sequences
- Covariance method - flame detection.
- Neural Network -produces accurate result as it uses temperature, smoke density and CO concentration

### Literature survey-2

The proposed system uses YCbCr colour spaces. Because YCbCr colour space separates luminance from chrominance, hence it is robust to changing illumination than other colour spaces like RGB and rgb (normalized RGB). The proposed method not only separates fire flame pixels but also separates high temperature fire centre pixels by taking in to account of statistical parameters of fire image in YCbCr colour space like mean and standard deviation

### Literature survey-3

This paper proposes a large-scale monitoring system and deep learning-based forest fire detection model that can detect forest fires from video frames captured by UAV drones. The proposed CNN model successfully detects forest fires with 97.29% accuracy.

## 2.2 REFERENCES

1) Author Name - Ms.Tamil Mathi, Dr.L.Latha - A SURVEY ON FOREST FIRE DETECTION

2) Author Name - Priyadarshini Hanamaraddi - A Literature Study on Image Processing for Forest Fire Detection

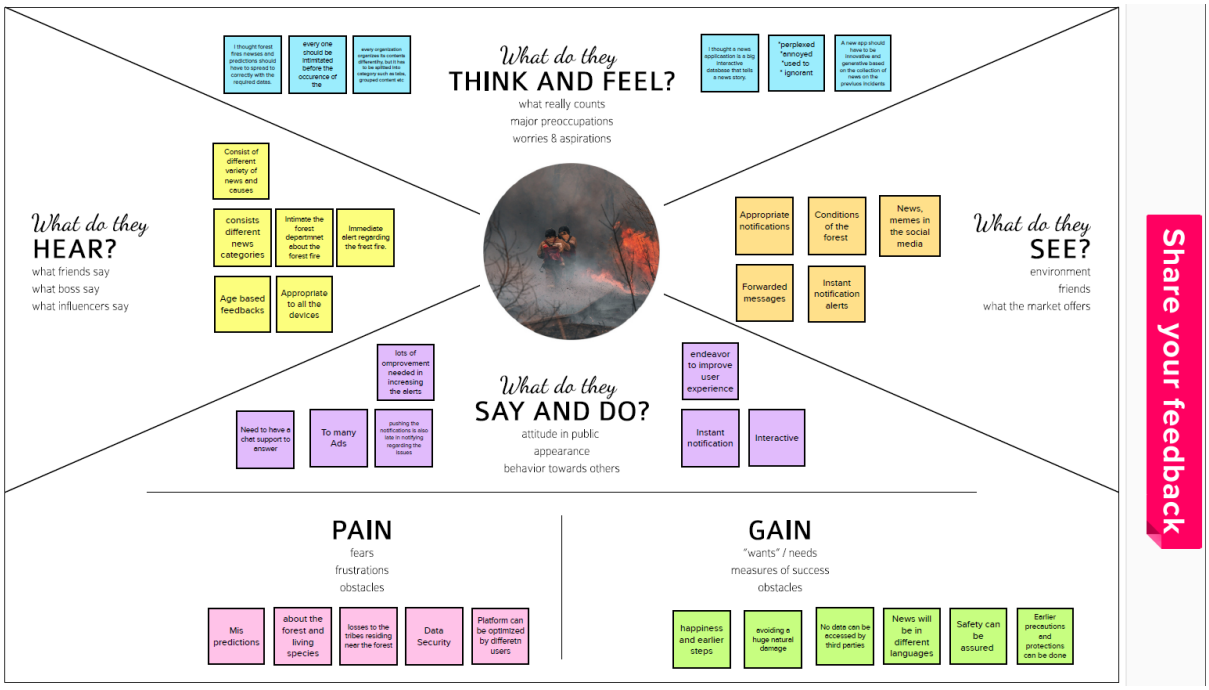
3) Author Name - Abhay Chopde , Ansh Magon , Shreyas Bhatkar - Forest Fire Detection and Prediction from image processing using RCNN

## 2.3 PROBLEM STATEMENT DEFINITION

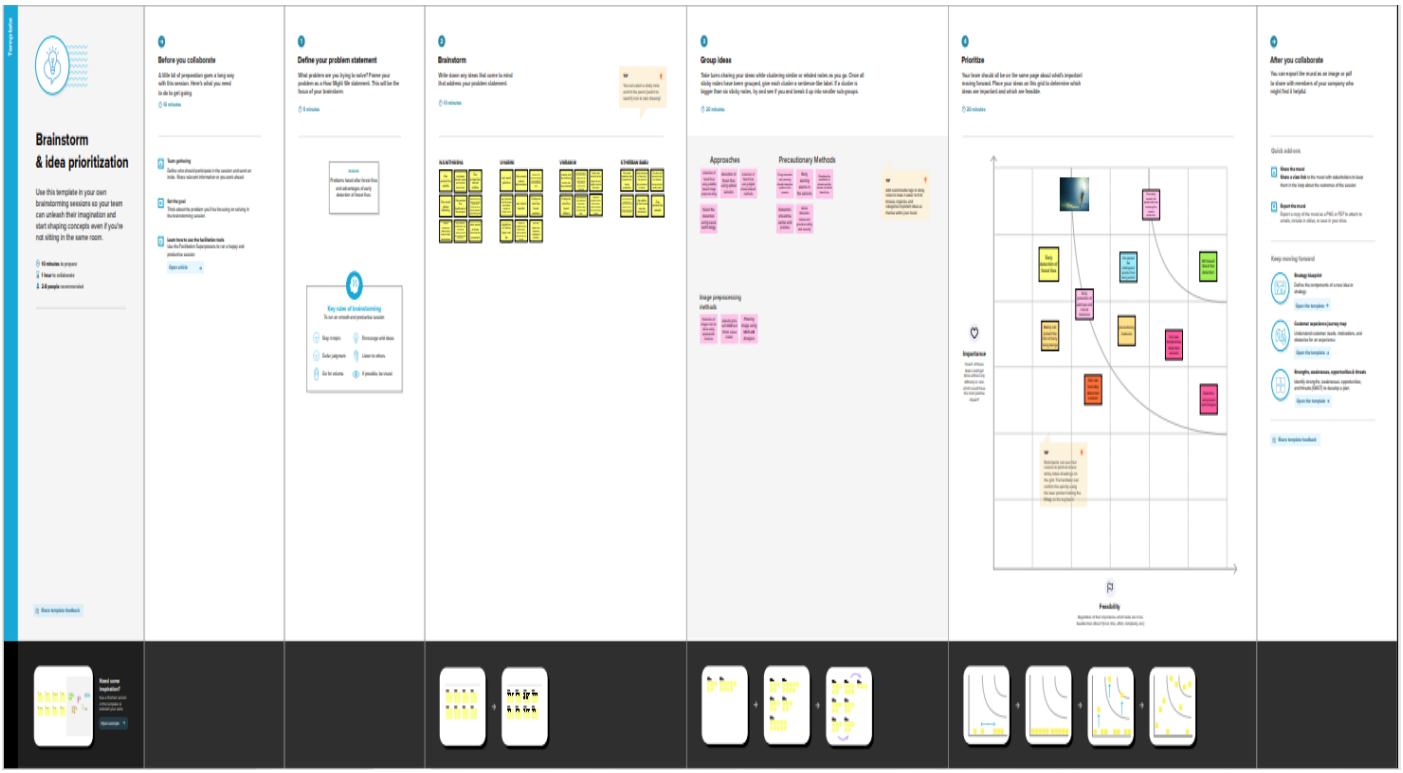
| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (problem to be solved) | Forest fires are one of the most vulnerable natural disaster. It affects thousand acres of land and natural resources, which also affects the thousands of living organisms and affects the livelihood of tribes and other forest workers. |
| 2. | Idea / Solution description | The earlier detection of forest fires requires 24/7 monitoring of the forest which is not possible manually. To avoid the delay and manual monitoring<br>➢ We can use image processing to monitor forest.<br>➢ Use computer vision methods for recognition and detection of smoke or fire, based on the still images or the video input from the drone cameras |
| 3. | Novelty / Uniqueness | ➢ Real time computer detects the forest fires at the earliest before being spread or to occur.<br>➢ Satellite images of forests will be processed for identifying the fire. |
| 4. | Social Impact / Customer Satisfaction | Early detection will helps to<br>➢ Prevent the loss of natural resources and wildlife.<br>➢ It can prevent the livelihood of the surroundings and the people getting affected after the occurrence of forest fires. |
| 5. | Business Model (Revenue Model) | ➢ The proposed method was implemented using the Python programming language on a Corei3 or greater.<br>➢ The prediction might be right and the can have high reliability. |
| 6. | Scalability of the Solution | ➢ In future we can include machine learning and robotics many other advance technologies to detect the smoke coming from the forest.<br>➢ Computer vision models enable land cover classification and smoke detection from satellite and ground cameras. |

# 3 IDEATION & PROPOSED SOLUTION

## 3.1 EMPATHY MAP CANVAS




## 3.2 IDEATION & BRAINSTORMING:

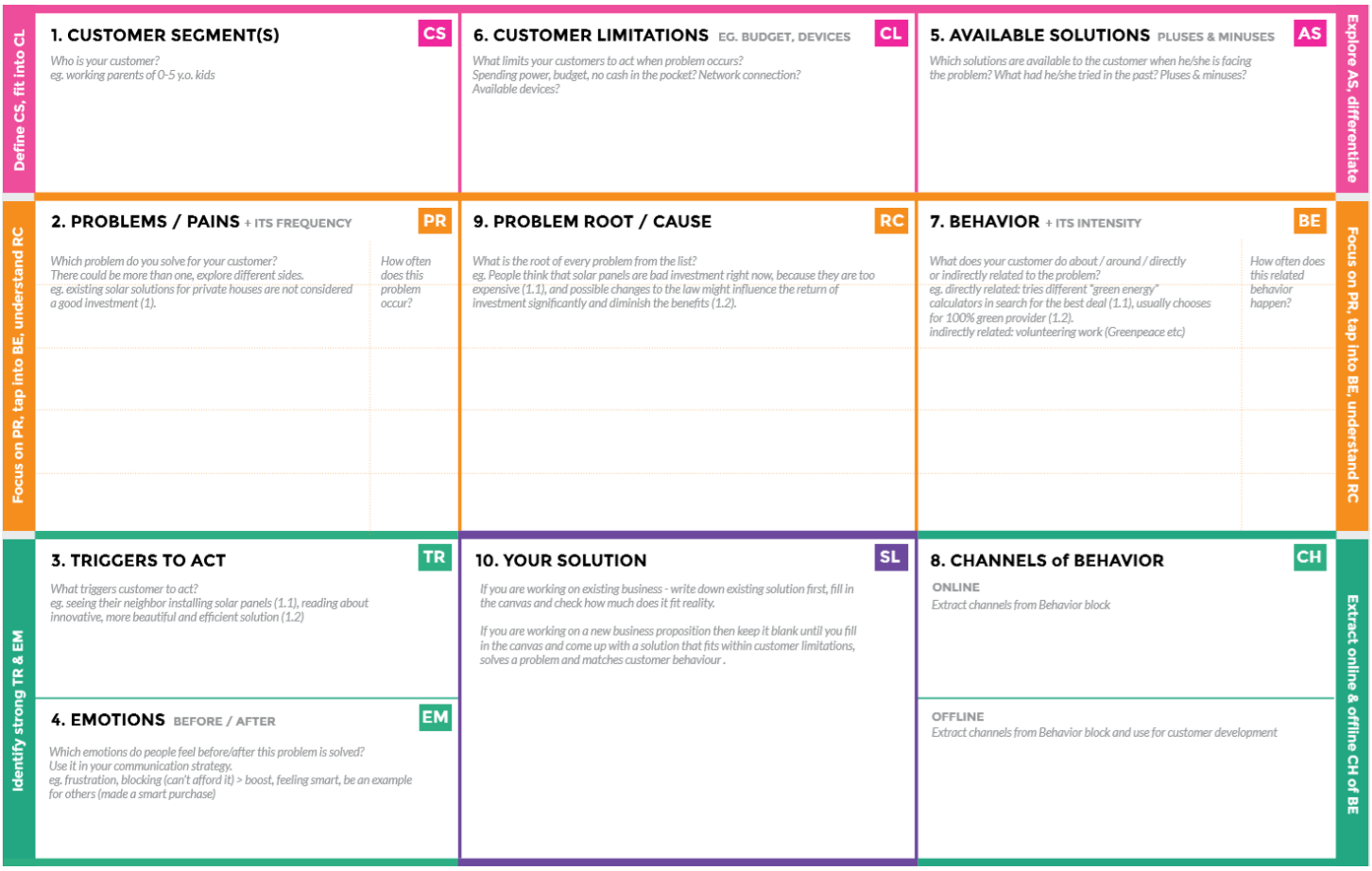


## 3.3 PROPOSED SOLUTION:

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

| S.No. | Parameter | Description |
|---|---|---|
| 1. | Problem Statement (problem to be solved) | Forest fires are one of the most vulnerable natural disaster. It affects thousand acres of land and natural resources, which also affects the thousands of living organisms and affects the livelihood of tribes and other forest workers. |
| 2. | Idea / Solution description | The earlier detection of forest fires requires 24/7 monitoring of the forest which is not possible manually. To avoid the delay and manual monitoring<br>➢ We can use image processing to monitor forest.<br>➢ Use computer vision methods for recognition and detection of smoke or fire, based on the still images or the video input from the drone cameras |
| 3. | Novelty / Uniqueness | ➢ Real time computer detects the forest fires at the earliest before being spread or to occur.<br>➢ Satellite images of forests will be processed for identifying the fire. |
| 4. | Social Impact / Customer Satisfaction | Early detection will helps to<br>➢ Prevent the loss of natural resources and wildlife.<br>➢ It can prevent the livelihood of the surroundings and the people getting affected after the occurrence of forest fires. |
| 5. | Business Model (Revenue Model) | ➢ The proposed method was implemented using the Python programming language on a Corei3 or greater.<br>➢ The prediction might be right and the can have high reliability. |
| 6. | Scalability of the Solution | ➢ In future we can include machine learning and robotics many other advance technologies to detect the smoke coming from the forest.<br>➢ Computer vision models enable land cover classification and smoke detection from satellite and ground cameras. |

## 3.4 PROBLEM SOLUTION FIT



# 4 REQUIREMENT ANALYSIS

## 4.1 FUNCTIONAL REQUIREMENT

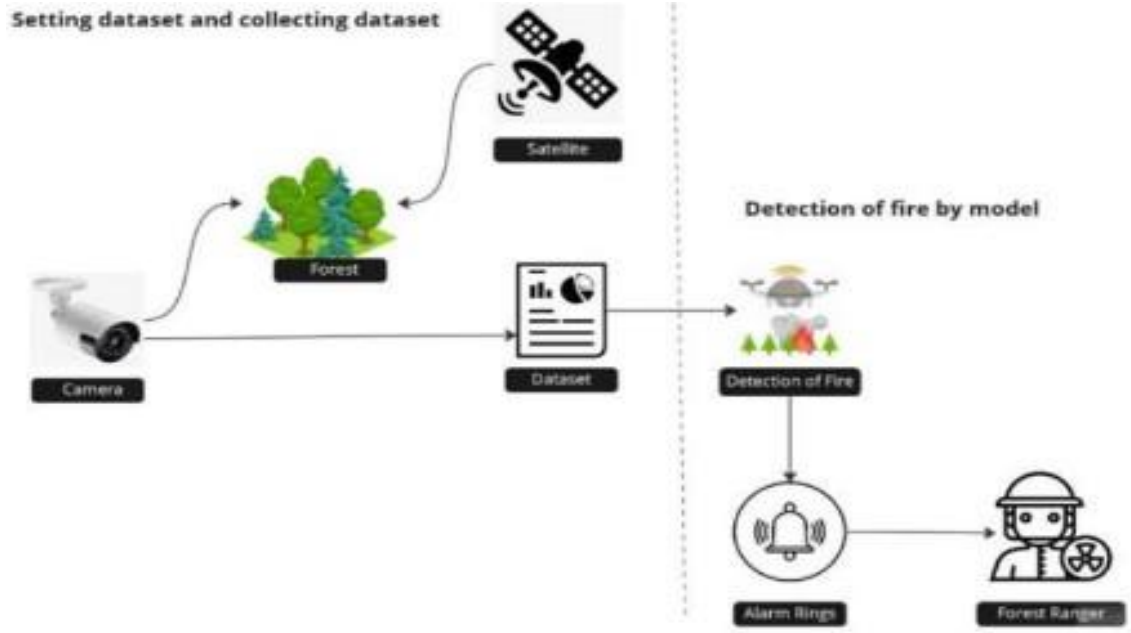
| FR. NO. | Functional Requirement | Sub Requirement (Story / Sub-Task) |
|---|---|---|
| FR-1 | User Registration | Registration through Form Registration through wildfire portal. |
| FR-2 | User Confirmation | Confirmation via Email Confirmation via OTP |
| FR-3 | Data Prediction | Scientists create computer models to predict wildfire potential under a range of potential climate futures. Using different projections of temperature and downfall, scientists predict where and when wildfires are likely to occur |

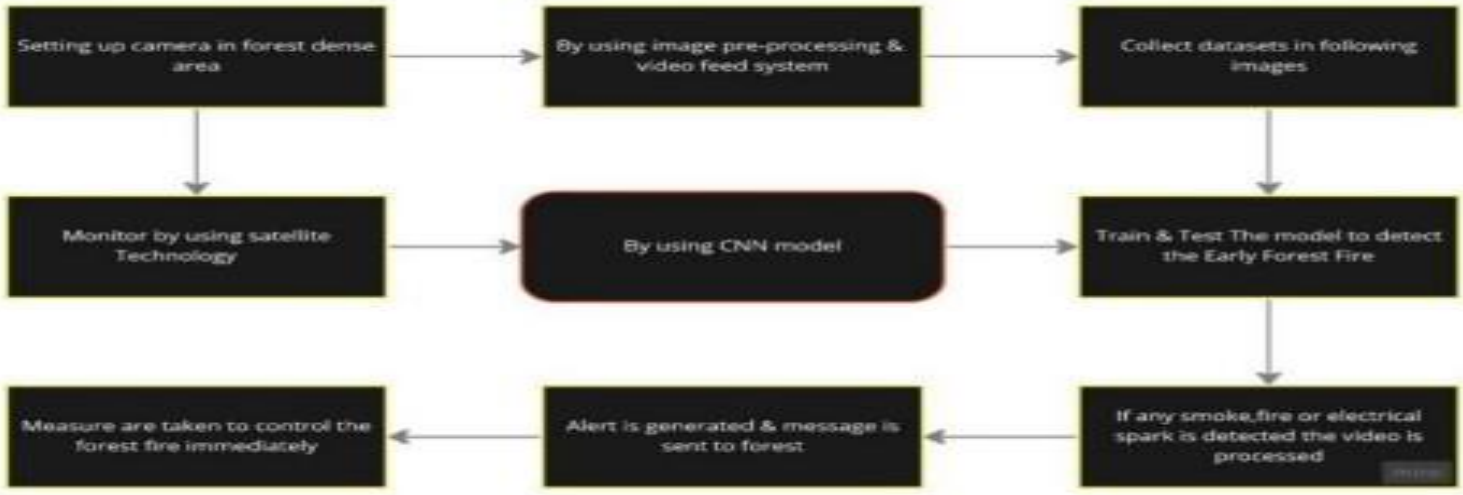
## 4.2 NON- FUNCTIONAL REQUIREMENTS

| FR. NO. | Non-Functional Requirement | Description |
|---|---|---|
| NFR-1 | Usability | Many methods have been proposed to detect forest fires, such as camera-based systems, WSN-based systems, and machine learning coating-based systems, with both positive and negative aspects and performance figures of detection. |
| NFR-2 | Protection | We have designed this project to secure the forest from wild fires. |
| NFR-3 | Performance | In the event of a fire, the primary objective of using drones is to gather situational consciousness, which can be used to direct the efforts of the firefighters in locating and controlling hot spots. Just like urban fires, forest fires to require monitoring so that firefighters know what they are dealing with. |

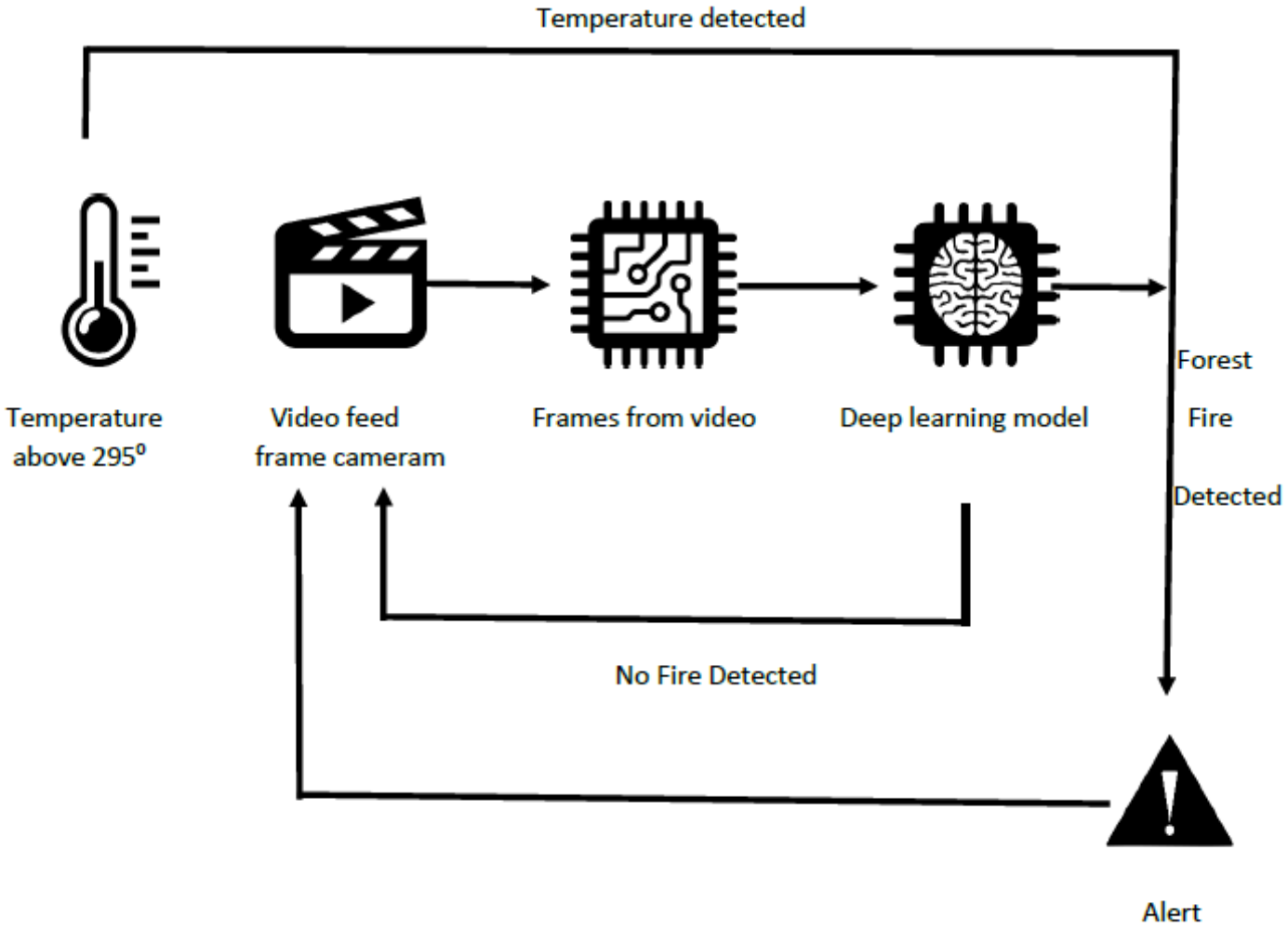
# 5 PROJECT DESIGN

## 5.1 DATAFLOW DIAGRAMS



### 5.2 SOLUTION & TECHNICAL ARCHITECTURE




# 6 PROJECT PLANNING & SCHEDULING

### 6.1 SPRINT PLANNING AND ESTIMATION

| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-1 | DATA COLLECTION | USN-1 | Data collected by sensors aboard orbiting satellites, carried aboard aircraft, or installedon the ground provide a wealth of data that can be used to assess conditions before a burn and track the movement of a wildfire innear real-time. | 10 | High | N.S.NITHIKSHA<br>V.HARINI<br>M.VIKRAM<br>S.THEEBAN BABU |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-1 | IMAGE PREPROCESSING | USN-2 | Image processing-Image processing technique automatically detect forest fires around the world by using infrared(IR) images sourced from satellites and CNN used for image recognition and tasks that involve the processing of pixel data. | 7 | Medium | N.S.NITHIKSHA<br>V.HARINI<br>M.VIKRAM<br>S.THEEBAN BABU |
| Sprint-2 | TRAINING AND TESTING | USN-3 | The model is trained for detecting the fire bytraining with real time work and the testing isdone according the accuracy of the model | 10 | high | N.S.NITHIKSHA<br>V.HARINI<br>M.VIKRAM<br>S.THEEBAN BABU |

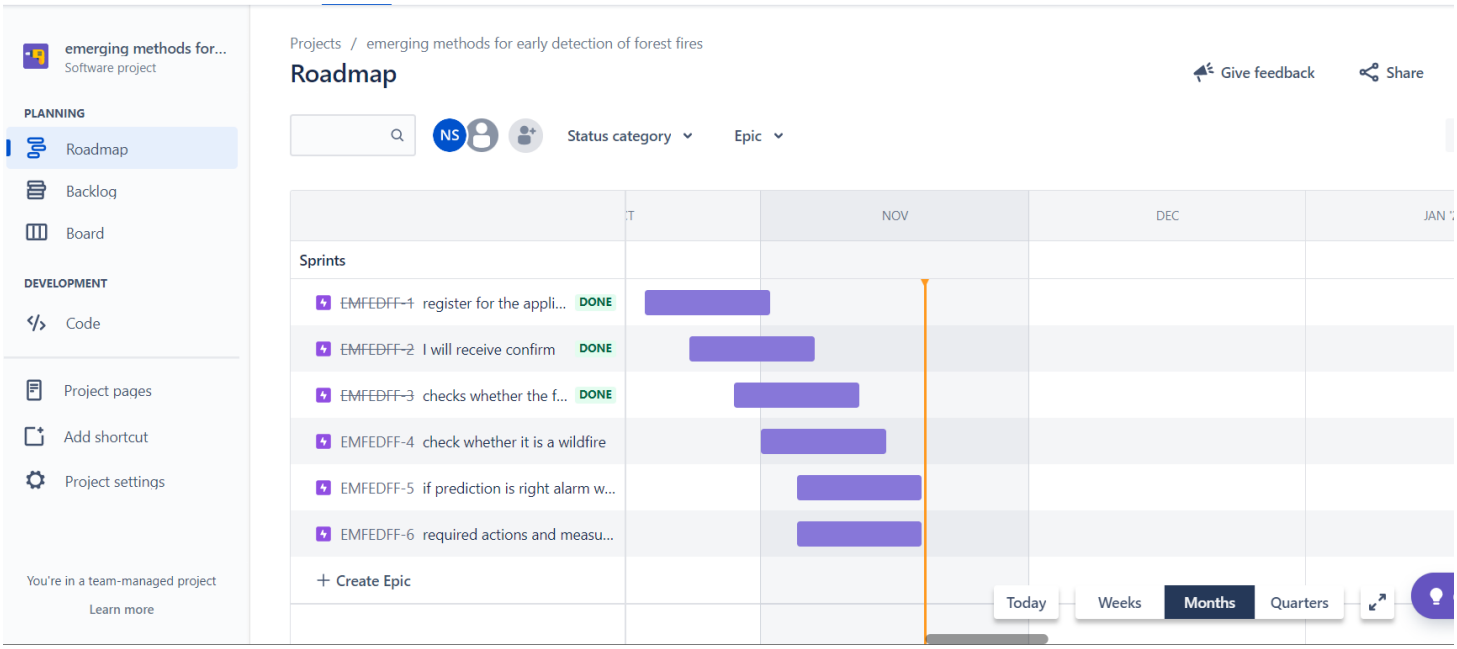
| Sprint | Functional Requirement (Epic) | User Story Number | User Story / Task | Story Points | Priority | Team Members |
|---|---|---|---|---|---|---|
| Sprint-3 | REVIEWING THEMODEL | USN-4 | The main task is to check that the model is efficient to work in real time to ensure there isno error in the model | 7 | Medium | N.S.NITHIKSHA<br>V.HARINI<br>M.VIKRAM<br>S.THEEBAN BABU |
| Sprint-4 | IMPLEMENTATION | USN-5 | After completing every step the model isimplemented on the forest and the quickresponses is collected from forest organization | 10 | High | N.S.NITHIKSHA<br>V.HARINI<br>M.VIKRAM<br>S.THEEBAN BABU |

## 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 | 27 Oct 2022 | 8 | 27 Oct 2022 |
| Sprint-2 | 20 | 6 Days | 31 Oct 2022 | 05 Nov 2022 | 7 | 08 Nov 2022 |

| | | | | | | |
|---|---|---|---|---|---|---|
| Sprint-3 | 20 | 6 Days | 07 Nov 2022 | 12 Nov 2022 | 8 | 16 Nov 2022 |
| Sprint-4 | 20 | 6 Days | 14 Nov 2022 | 19 Nov 2022 | 7 | 20 Nov 2022 |
| | | | | | | |
| | | | | | | |

### 6.3 REPORT FROM JIRA




# 7 CODING & SOLUTIONING (EXPLAIN THE FEATURES ADDED IN THE PROJECT ALONG WITH CODE)

### 7.1 FEATURE 1

```
import tensorflow as tf
import numpy as np
from tensorflow import keras
import os
import cv2
from tensorflow.keras.preprocessing.image import
    ImageDataGenerator
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
```

```
train = ImageDataGenerator(rescale=1/255)10 test =
    ImageDataGenerator(rescale=1/255) 11train_dataset =
train.flow_from_directory("/content/drive/MyDrive/Dataset/train_
    set",target_size=(150,150),batch_size = 32,class_mode =
    'binary')

test_dataset = test.flow_from_directory("/content/drive/MyDrive/Dataset/test_set
    ",target_size=(150,150),batch_size =32,class_mode = 'binary')
test_dataset.class_indices

model = keras.Sequential()
model.add(keras.layers.Conv2D(32,(3,3),activation='relu',input_shape=(150,150,3)))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(64,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(128,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(128,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Flatten())
model.add(keras.layers.Dense(512,activation='relu'))
model.add(keras.layers.Dense(1,activation='sigmoid'))
model.summary()
model.compile(optimizer='adam',loss='binary_crossentropy',metrics
   =['accuracy'])

r = model.fit(train_dataset,epochs = 10,validation_data =test_dataset)
model.save("forest1.h5")
predictions = model.predict(test_dataset)31predictions =
np.round(predictions)
predictions
print(len(predictions))
import matplotlib.pyplot as plt
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.plot(r.history['accuracy'], label='accuracy')
plt.legend()
plt.plot(r.history['accuracy'],label='accuracy')
plt.plot(r.history['val_accuracy'], label='val_accuracy')
plt.legend()
def predictImage(filename):
        img1 = image.load_img(filename,target_size=(150,150))
        plt.imshow(img1)
        Y = image.img_to_array(img1)
        X = np.expand_dims(Y,axis=0)
        val = model.predict(X)
```

```
        print(val)
        if val == 1:
                plt.xlabel("Fire")
        elif val == 0:
                plt.xlabel("No Fire")
predictImage(r"C:\Users\ELCOT\Desktop\Python\new\Dataset\test_set
\with fire\599857.jpg")
predictImage(r"C:\Users\ELCOT\Desktop\Python\new\Dataset\test_set
\forest\01_NeilBurnell_Mystical_photoverticall.jpg")
predictImage(r"C:\Users\ELCOT\Desktop\Python\new\Dataset\train_set\with fire\with fire
(2).jpg")
```

## 7.2 FEATURE 2

## FLASK INTEGRATION

```
from____future_____import division, print_function
import os
import numpy as np
import tensorflow as tf
from tensorflow.keras.preprocessing import image
from tensorflow.keras.models import load_model
from flask import Flask, request, render_template
from werkzeug.utils import secure_filename
from twilio.rest import Client10
global graph
#graph=tf.get_default_graph()13#
Define a flask app
app = Flask(________name___)
model = load_model('forest1.h5')16


@app.route('/', methods=['GET'])
def index():
        # Main page
        return render_template('digital.html')23

24

@app.route('/predict', methods=['GET',
'POST'])26 def upload():
        if request.method == 'POST':
                # Get the file from post request
                f = request.files['image']30
                # Save the file to ./uploads
                basepath = os.path.dirname(_____________file_____)
                file_path = os.path.join(
                        basepath, 'uploads', secure_filename(f.filename))
                f.save(file_path)
```

```
            img1 = image.load_img(file_path,target_size=(150,150))

            y = image.img_to_array(img1)
            x= np.expand_dims(y, axis=0)
            val = model.predict(x)
            print(val)
            if val == 1:
                    send_message()
                    result ="Fire"
            elif val == 0:
                    result = "No Fire"
            return result48
def send_message():
            client = Client(account_sid, auth_token)
            message = client.messages.create(
                    body="Forest Fire detected , Stay safe!!!",
                    )
            print(message.sid)
            print("Fire Detected")
            print("SMS Sent!")57
if_____name___== '_____main___':
        app.run(threaded = False60
```

# 8 TESTING

## 8.1 TEST CASES

| Test case ID | Feature Type | Component | Test Scenario | Steps To Execute | Test Data | Expected Result | Actual Result | Status | BUG ID | Executed By |
|---|---|---|---|---|---|---|---|---|---|---|
| HomePage_ TC_O O1 | UI | HomePage | Display the Emerging Methods of Forest Fires | Enter URL(http://127.0.0.1:5000) | http://127.0.0.1:5000 | Displaying the Home Page | Home Page displayed | **Pass** | - | N.S.Nithiksha V.Harini |
| HomePage_ TC_O O2 | UI | HomePage | Displayed the Prediction Forest Fires | 1.Enter URL(http://127.0.0.1:5000) 2.Content of the Home Page should be Displayed | http://127.0.0.1:5000 | Displaying the contentof Home page | Content of Home page is displayed | **Pass** | - | N.S.Nithiksha S.Theeban babu |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| HomePage_ TC_O O3 | Functional | Home | Checks whether the Drop the Image Here! Button is visible | 1. Enter URL(http://127.0.0.1:5000) 2. Content of the Home Page should be Displayed 3.Click Drop the Image Here! Button | http://127. 0.0.1:5000 | Displays the Button | Drop the Image Here! Button is pops up. | Pass | - | S.Theeban V.Harini |
| PredictiedPa ge_T C_OO4 | Functional | Predicted page | Display the Prediction Page and Choose Image Button | 1. Enter URL(http://127.0.0.1:5000) 2. Content of the Home Page should be Displayed 3.Click Drop the Image Here! Button 4.Click Choose Image | http://127. 0.0.1:5000 | Displays the Prediction Page and the Choose Image Button | Predicti on page displaye d. Choose image button was clicked. | Pass | - | M.Vikram N.S.Nithiksha |
| PredictedPa ge_TC _OO4 | Functional | Predicted page | Select the Image and Click the Predict Button | 1.Enter URL(http://127.0.0.1:5000) 2.Content of the Home Page should be Displayed 3.Click Drop the Image Here! Button 4.Click ChooseImage 5.Choose the Image and Click Predict | http://127. 0.0.1:5000 | Displays e the selected Image | Displays the Selected Imag | Pass | - | N.S.Nithiksha S.Theeban |

## 8.2 USER ACCEPTANCE TESTING

## DEFECT ANALYSIS

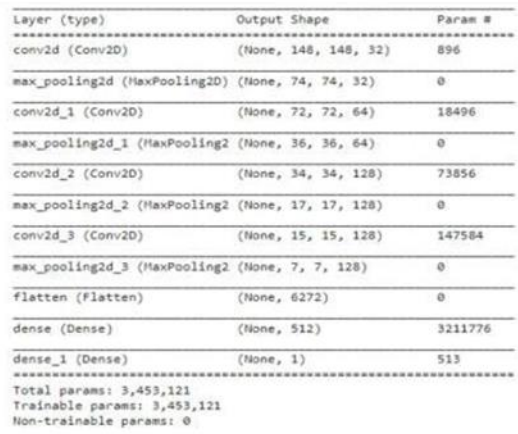
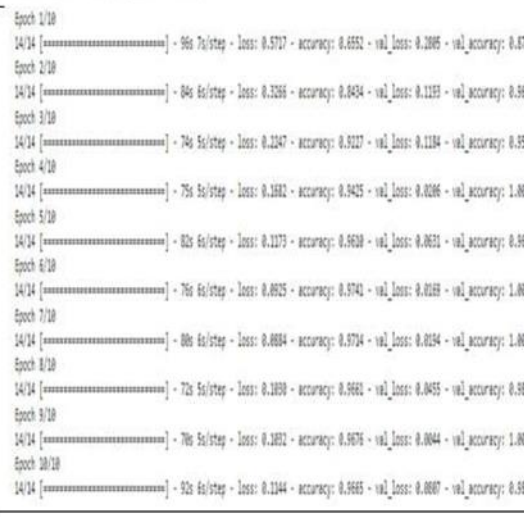
| Resolution | severity 1 | severity 2 | severity 3 | badness 4 | subtotal |
|---|---|---|---|---|---|
| By Design | 1 | 1 | 2 | 0 | 4 |
| Duplicate | 0 | 0 | 0 | 0 | 0 |
| External | 0 | 0 | 2 | 1 | 3 |
| Fixed | 4 | 2 | 4 | 1 | 11 |
| Not Reproduced | 0 | 0 | 0 | 0 | 0 |
| Skipped | 0 | 0 | 1 | 1 | 2 |
| Won't Fix | 0 | 0 | 0 | 1 | 1 |
| Totals | 5 | 3 | 9 | 4 | 21 |

**TEST CASE ANALYSIS**

| Section | Total Cases | Not tested | Fail | Pass |
|---|---|---|---|---|
| Client Application | 10 | 0 | 0 | 10 |
| Security | 2 | 0 | 0 | 2 |
| Performance | 2 | 0 | 0 | 2 |
| Exception Reporting | 2 | 0 | 0 | 2 |
| Final Report Output | 3 | 0 | 0 | 3 |

# 9 RESULTS

## 9.1 PERFORMANCE METRICES

| S. No | Parameter | Values | Screenshot |
|---|---|---|---|
| 1. | Model Summary | 3,453,121 |  |
| 2. | Accuracy | Training Accuracy - 0.9665<br>Validation Accuracy -0.9833 |  |

# 10 ADVANTAGES & DISADVANTAGES

## ADVANTAGES:

- The proposed system detects the forest fire at a faster rate compared to existing system. Ithas enhanced data collection feature.
- The major aspect is that it reduces false alarm and also has accuracy due to varioussensors present.
- It minimises the human effort as it works automatically. This is very low-cost due towhich can be easily accessed.
- The main objective of our project is to receive an alert message through an app to therespective user.

## DISADVANTAGE:

- The electrical interference diminishes the potency of radio receiver.
- The main drawback is that it has less coverage range areas.

# 11 CONCLUSION

This type of system is the first of its kind to ensure no further damage is then to forests when there is fire breakout and instantly a message is sent to the user through the App.Immediate response or early warning to a fire breakout is mostly the only ways to avoid lossesand biology, cultural heritage damages to a great extent. Therefore, the most important goals in fire surveillance are quick and authentic detection of fire. It is so much easier to suppress fire while it is in its early stages. info about progress of fire is highly valuable for managing fire.

During all its stages. Based on this data the firefighting staff can be guided on target to block firebefore it reaches cultural heritage sites and to suppress it quickly by utilise required firefighting equipment and vehicles. With further research and invention, this project can be implemented invarious forest areas so that we can save our forests and maintain great environs.

# 12 FUTURE SCOPE

This project is far from complete and there is a lot of room for betterment. Some of thebetterment that can be made to this project are as follows:

Additional pump can be added so that it automatically sends water when there is a fire breakout. Also industrial sensors can be used for better ranging and accuracy.

This project has endless potential and can always be enhanced to become better. Enforce this concept in the real world will benefit several industries and reduce the workload on many workers, enhancing overall work efficiency.

# 13 APPENDIX

SOURCE CODE

```
import tensorflow as tf
import numpy as np
from tensorflow import keras
import os
import cv2
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.preprocessing import image
import matplotlib.pyplot as plt
```

```
train = ImageDataGenerator(rescale=1/255)
test = ImageDataGenerator(rescale=1/255)

train_dataset =
  train.flow_from_directory("/content/drive/MyDrive/Dataset/train_set
  ",
                                                  target_size=(150,150),
                                                  batch_size = 32,
                                                  class_mode = 'binary')

test_dataset =
  test.flow_from_directory("/content/drive/MyDrive/Dataset/test_set",
                                                  target_size=(150,150),
                                                  batch_size =32,
                                                  class_mode = 'binary')
```

```
test_dataset.class_indices
```

```
model = keras.Sequential()
model.add(keras.layers.Conv2D(32,(3,3),activation='relu',input_shape=(150,150,3)))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(64,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(128,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Conv2D(128,(3,3),activation='relu'))
model.add(keras.layers.MaxPool2D(2,2))
model.add(keras.layers.Flatten())
model.add(keras.layers.Dense(512,activation='relu'))
model.add(keras.layers.Dense(1,activation='sigmoid'))
```

```
model.compile(optimizer='adam',loss='binary_crossentropy',metrics=['accuracy'])
```

```
r = model.fit(train_dataset,
          epochs = 10,
          validation_data = test_dataset)
```

```
predictions
print(len(predictions))
```

```
import matplotlib.pyplot as plt
plt.plot(r.history['loss'], label='loss')
plt.plot(r.history['val_loss'], label='val_loss')
plt.plot(r.history['acc'], label='acc')
plt.legend()
```

```
plt.plot(r.history['acc'], label='acc')
plt.plot(r.history['val_acc'], label='val_acc')
plt.legend()
```

```
def predictImage(filename):
        img1 = image.load_img(filename,target_size=(150,150))
        plt.imshow(img1)
        Y = image.img_to_array(img1)
        X = np.expand_dims(Y,axis=0)
        val = model.predict(X)
        print(val)
        if val == 1:
                plt.xlabel("Fire",fontsize=30)
        elif val == 0:
                plt.xlabel("No Fire",fontsize=30)
```

```
predictImage("/content/drive/MyDrive/Dataset/test_set/
with fire/19464620_401.jpg")
```

```
predictImage('/content/drive/MyDrive/Dataset/test_set/forest/0.480072
00_1530881924_final_forest.jpg')
```

```
predictImage('/content/drive/MyDrive/Dataset/train_set/with
fire/withfire (101).jpg')
```

```
predictImage('/content/drive/MyDrive/Dataset/test_set/forest/0.480072
00_1530881924_final_forest.jpg')
```

```
predictImage('/content/drive/MyDrive/Dataset/test_set/
with fire/19464620_401.jpg')
```

# GITHUB & PROJECT DEMO LINK

GITHUB LINK

IBM-EPBL/**IBM-Project-31285-1660198652**

# PROJECT DEMO LINK

**https://drive.google.com/file/d/1-S2I2zGDTYGBLFLOBwzZMyZdMCqICPJU/view?usp=share_link**