TEAM ID: PNT2022TMID10687

EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRES

TEAM LEADER: RAGAPRIYA. N (4211191021076)

TEAM MEMBER:

1. RAMANI. R (4211191021080)
2. POOJA SRI. R (4211191021071)
3. PRINCY CELSIA. F (4211191021074)

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

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. 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.

1.2 PURPOSE

In the past, watching towers and satellite photos were used to detect fires. Satellites gather photographs and relay them to the monitoring authority, who determines whether or not a fire is presently based on the images. But given the possibility that the fire had spread to such huge areas and caused such damage, this approach was extraordinarily slow. Significant damage before the rescue crew arrived. The watching tower method required a man to always be present, who would watch the area and report if there was a fire. This approach was also delayed because the fire might already have moved into the interior areas of the forest by the time the man learned about it. Given that some regions, particularly forest areas, are vast, it is nearly impossible to place a man in every area of the forest from where they can monitor the forest area. As a result, both these methods—watching towers and satellite images—failed to identify fire as soon as feasible to lessen the damage it caused. Issues in detecting fires. The edge is set, and if the worth is greater than the edge, the object is deemed to be a fire; otherwise, it is not. As a result, numerous researchers eliminated this issue by utilizing machine learning approaches. Nodes are connected using cables in traditional systems. Cables were primarily copper. However, copper wire is expensive or the cable may be defective in the middle. This problem has been solved by wireless sensor networks. So as technology advances, researchers find efficient ways to detect wildfires using a wireless sensor network. Conveying sensor hubs in the timberland regions means placing sensors in every part of the forest and mostly in the prone areas where the risk of catching fire is more. With the use of wireless sensor networks, now it is easy to detect fire in large areas as soon as possible.

CHAPTER-2

LITERATURE SURVEY

2.1 EXISTING SYSTEM

- Existing systems use electronic sensors to detect fire or smoke. The change in temperature indicates the presence of fire or smoke in a region which can be detected by the sensors using radiation heat.
- In research done by Zhang et al. (2009), Pirbhulal et al. (2017), and Alahi et al. (2017) an ad hoc network using cluster topology for forest fire forecasting model was used to predict fire prone areas. It was concluded that WSNs have greater advantages.
- In another research done by Demin et al. (2014), sensors were deployed and the weather data were collected. This data was used to calculate and prevent forest fires. In these researches, there was no real-time forest fire monitoring, only the data were collected and fire prone areas were predicted.

2.2 REFERENCES

1. **Publish:** Published on 2020

   Title: Early Forest Fire Detection Systems Using Optical Remote Sensing

   Description: To detect forest fires early, the proper categorization of fire and fast response from the firefighting departments. The fire detection is based on a platform that uses.Unmanned Aerial Vehicles (UAVs) which constantly patrol over potentially threatened by fire areas. The UAVs utilize the benefits from Artificial Intelligence (AI). This allows to use computer vision methods for recognition and detection of smoke or fire, based on images or video input from the drone cameras. From this journal, we use drone cameras and UAVs, because it patrols the forest always.

2. **Publish:** Published on 2020

   Title: Early Forest Fire Detection Systems Using Optical Remote Sensing

   Description: To fight forest fires occurring throughout the year with an increasing intensity in the summer and autumn periods. Detection methods that use optical sensors or RGBcamerascombine features that are related to the physical properties of flame and smoke, such as color,motion, spectral, spatial, temporal, and texture characteristics. From this journal, we use modern optical sensor networks which are known for their long range communication capabilities and extremely suitable for sensor and telemetry applications.

3. Publish: Published on 2018 IEEE.

Title: Developing a real-time and automatic early warning system for forest fire.

Description: To detect forest fires causing by climatic conditions and also caused by human. The method using here is making use of stand-alone boxes which are deployed throughout the forest. Those boxes contain different sensors and a radio module to transmit data received from these sensors. Each sensor will be tested in individually and XBee modules are configured and paired using XCTU Software. From this journal, we use Software solutions which are used for implementing microcontroller kits and to simulate and designing circuit boards.

4. Publish: Published on 2018 IEEE.

Title: Early FireDetection System using wireless sensor networks.

Description: To detect fires from huge cause of forests The hierarchical architecture of Wireless Sensor Networks is most efficient and extensible for dense networks which simplifies the management of the forest as well as the communication and the localization of fire and sensors. From this journal, we use cluster heads as landmark for the rest of sensor for localization in order to define their GPS coordinates according to the cluster head's coordinate.

5. Publish: Published 2018 IEEE

Title: Automatic Early Forest fire Detection based Gaussian Mixture Model.

Description: To avoid the huge damage of forest caused by fires. Based on the slow spread of smoke, firstly a time delay parameter improves Gaussian mixture model for extracting candidate smoke regions. Then, two motion features of smoke, the rate of area change and motion style are used to select smoke regions from the candidate regions. From this journal, we use Gaussian mixture model. Because it can reconstruct background with the advantages of small storage space, adaptive learning and good noise toleration.

LITERATURE SURVEY

1) Tanase, M.A.; Aponte, C.; Mermoz, S.; Bouvet, A.; Le Toan, T.; Heurich, M. Detection of windthrows and insect outbreaks by L-band SAR: A case study in the Bavarian Forest National Park. Remote Sens. Environ. 2018, 209, 700–71

2) Bu, F.; Gharajeh, M.S. Intelligent and vision-based fire detection systems: A survey. Image Vis. Comput. 2019, 91, 103803.

3) Muhammad, K.; Ahmad, J.; Mehmood, I.; Rho, S.; Baik, S.W. Convolutional neural networks based fire detection in surveillance videos. IEEE Access 2018, 6, 18174–18183. [CrossRef]

4) Shen, D.; Chen, X.; Nguyen, M.; Yan, W.Q. Flame detection using deep learning. In Proceedings of the 2018 4th International Conference on Control, Automation and Robotics (ICCAR), Auckland, New Zealand, 20–23 April 2018; pp. 416–420.
5) Wickramasinghe, C.; Wallace, L.; Reinke, K.; Jones, S. Intercomparison of Himawari-8 AHIFSA with MODIS and VIIRS active fire products. Int. J. Dig. Earth 2018

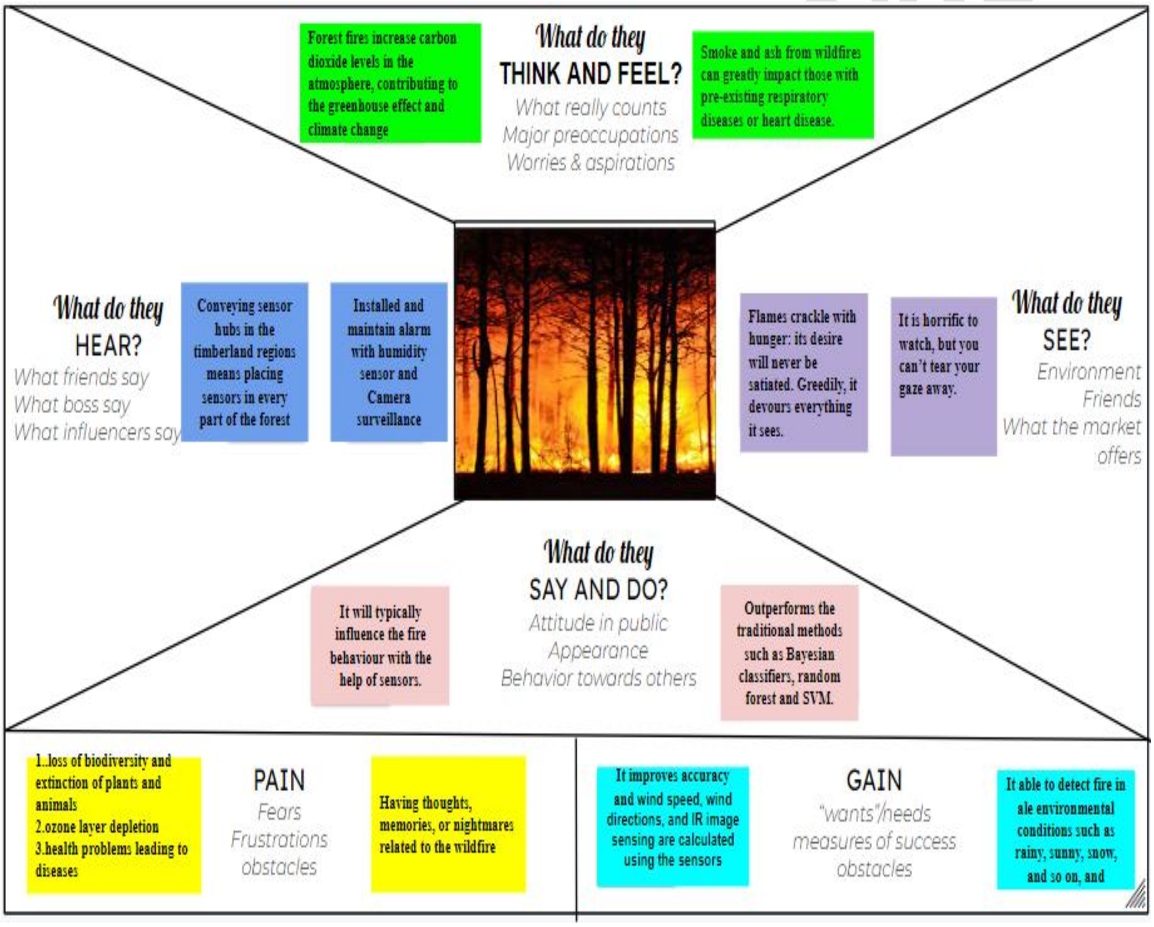
2.3 PROBLEM STATEMENT DEFINITION

Watching towers and satellite images have both been used in the past to find fires. Satellites take pictures and send them to the monitoring agency, which uses the pictures to decide whether or not there is a fire right now. This strategy, however, was incredibly slow given the likelihood that the fire had possibly spread to such vast areas and caused such damage. Before the rescue team arrived, there was a lot of damage. The man's decision to take this approach was also put off because by the time he learned about the fire, it might have already spread to the forest's interior. It is almost impossible to put a man in every area of the forest from which they can monitor the forest area because some regions, especially forest areas, are very large. As result, many researchers used machine learning techniques to solve this problem.

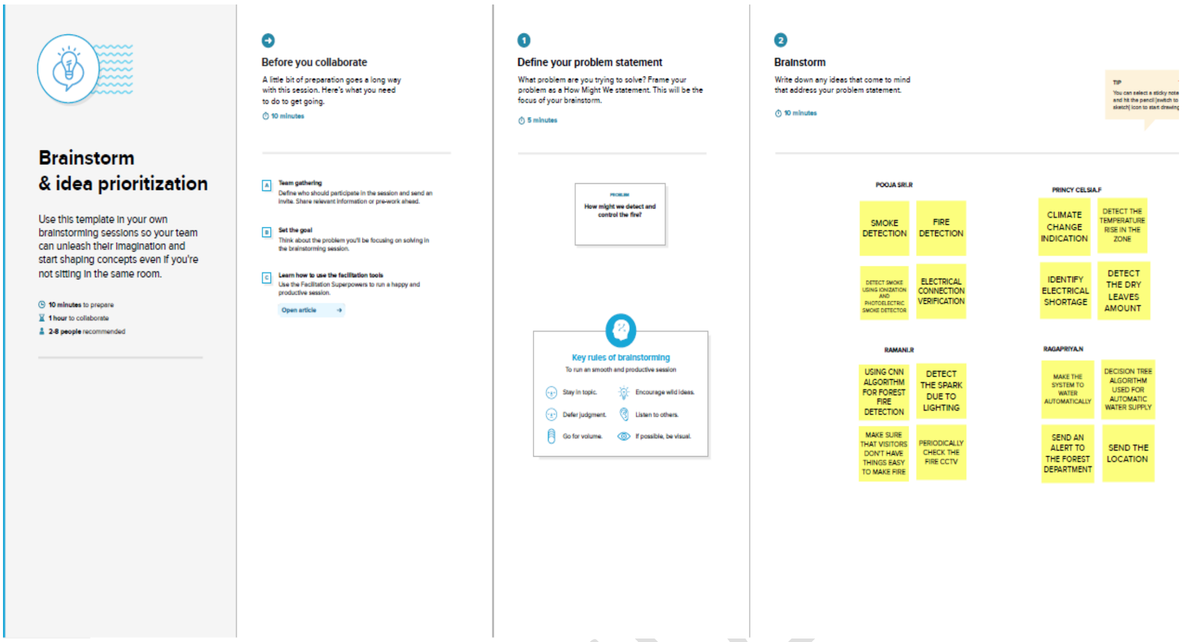
CHAPTER-3
IDEATION & PROPOSED SOLUTION

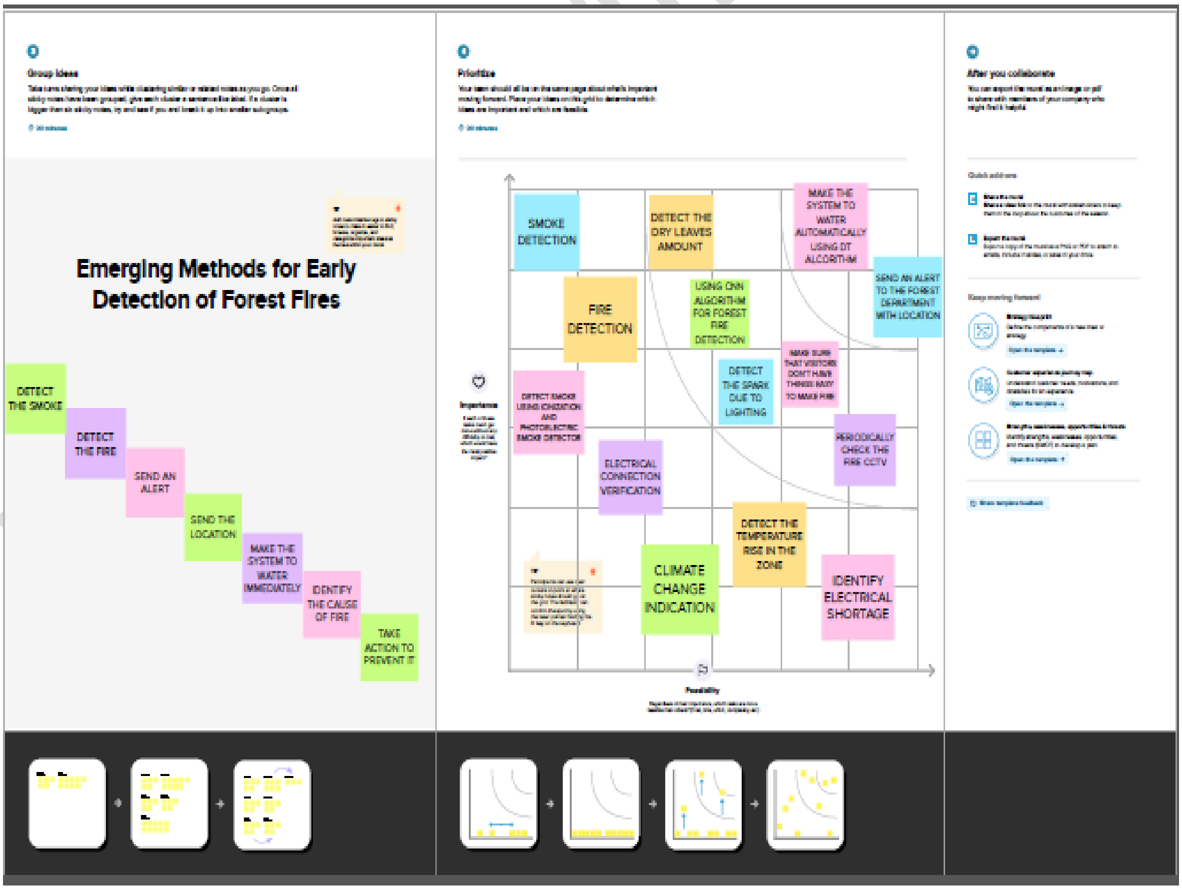
3.1 EMPATHY MAP CANVAS

The empathy map represents a principal user and helps teams better understand their motivations, concerns, and user experience. Perceptive engagement can be considered the pinnacle of empathic skill, because it combines your capacity to sense and accurately identify the emotions of others, regulate your own emotions, take the perspective of others, focus on them with care and concern, and then do something skilful based upon your perceptions. It is used for better understanding of the user, distilled information in one visual reference.



3.2 IDEATION & BRAINSTORMING





3.3 PROPOSED SOLUTION

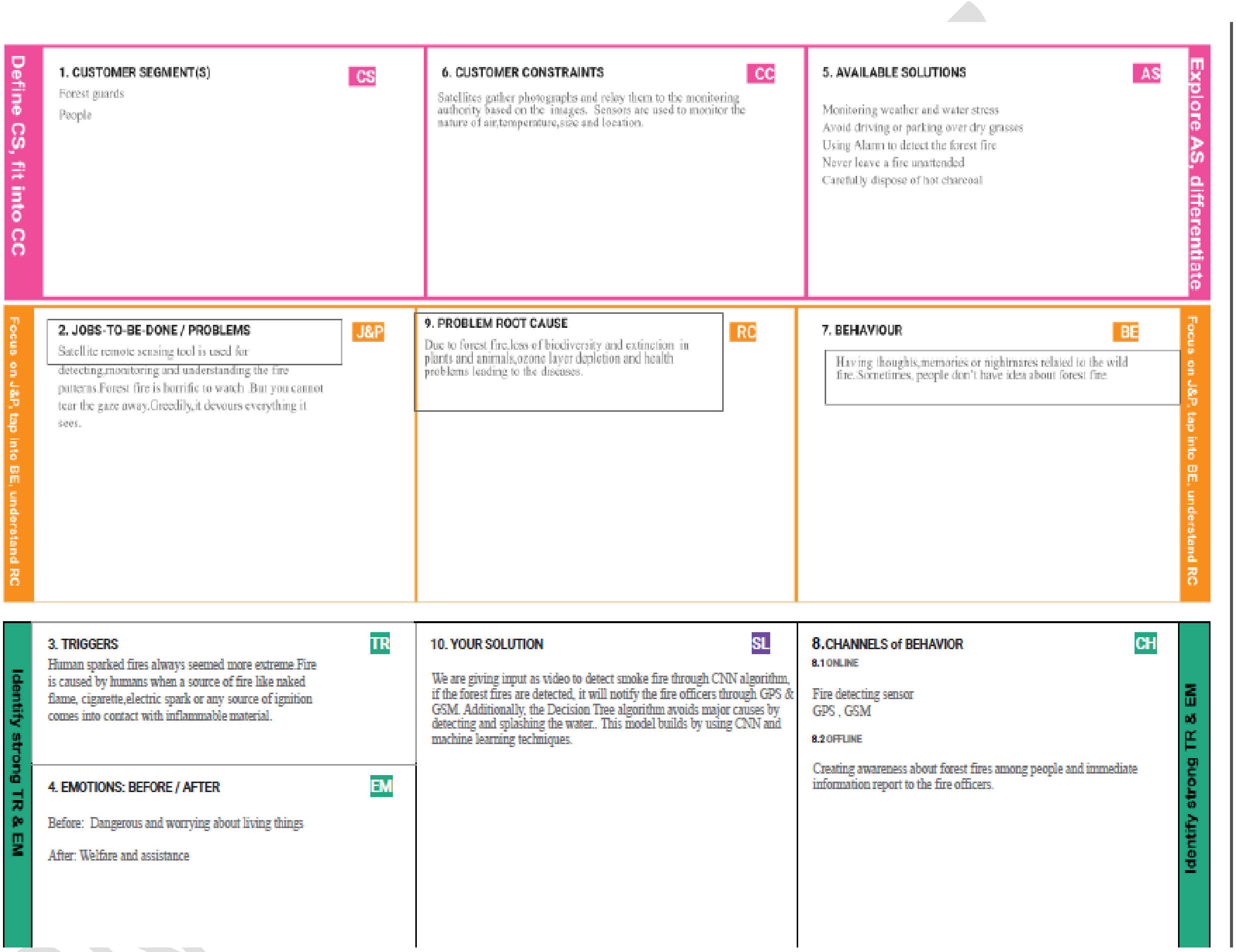
Forest fires pose a serious threat to the environment because they harm the economy, the ecosystem, and put people in danger. Dangerous wildfires have destroyed more than 9 million acres of land. In a sparsely populated forest area, it is challenging to predict and detect forest fires, and it is even more challenging if the prediction is made using ground-based techniques like camera or video-based approaches. The goal of the project is to create a kind of alarm system to detect the forest fire before or after the fire in order to enforce certain preventive measures. This can be achieved by using the Deep Learning and CNN model with the cumulative effect of the Artificial Intelligence technology. Due to their dependability and effectiveness, satellites can be a valuable source of data both before and during the Fire.

PARAMETERS:

- **Problem Statement (Problem to be solved):** A forest fire risk prediction algorithm, based on support vector machines, is presented. The algorithm depends on previous weather conditions in order to predict the fire hazard level of a day.
- **Idea / Solution description:** 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.
- **Novelty / Uniqueness:** Real-time computer programs detect forest fires earliest before they spread to larger areas.
- **Impact on society:** Blocked roads and railway lines, electricity, mobile and land telephone lines cut, destruction of homes and industries.
- **Business Model (Revenue Model):** The proposed method was implemented using the Python programming language on a Core i3 or greater (CPU and 4GB RAM.)
- **Scalability of the Solution:** Computer vision models enable land cover classification and smoke detection from satellite and ground cameras.

3.4 PROBLEM SOLUTION FIT

Project fit solution provides an appropriate solution for forest fire prediction analysis. It can strategize, plan, budget, implement, and manage the project.



CHAPTER-4

REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

- **Image Processing:** Transforming an image into a digital form.
- **Fire Detection**: The sense of one or more of the products or phenomena resulting from fire, such as smoke, heat, infrared and/or ultraviolet light radiation, or gas.
- **Smoke Detection**: A device that senses smoke, typically as an indicator of fire.
- **Alarm Generation:** An audible and/or visible means of indicating to the operator an equipment malfunction, process deviation, or abnormal condition.
- **Message Generation:** The assessment of an advertisement's effectiveness after it has been used.
- **Automatic Water Supply:** A watering system capable of being set to turn on and off automatically at pre-determined times, without human intervention.

4.2 NON-FUNCTIONAL REQUIREMENTS

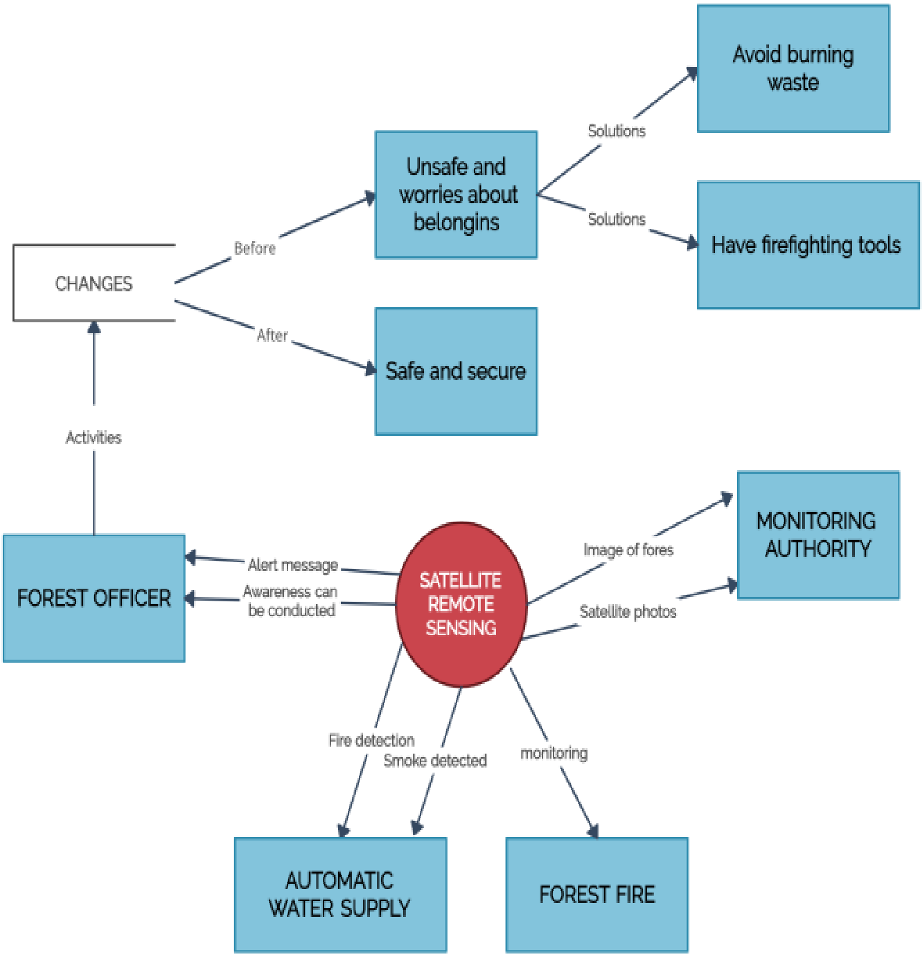
- **Usability:** Limit the emission of toxic products created by combustion, as well as global-warming gases produced by the fire itself.
- **Security:** It receives the signals from the detectors and manages and centralizes the entire response to the fire.
- **Reliability:** Important to ensure that safety-critical systems
- **Performance:** The formal analysis of the desired function of fire detection systems, to identify the performance parameters of fire detection systems.
- **Available:** Aspirating smoke detector generally consists of a central detection unit that sucks air through a network of pipes to detect smoke.
- **Scalability:** Better sensitivity than fire surveillance based on imaging.

CHAPTER-5

PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

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



5.2 SOLUTION & TECHNICAL ARCHITECTURE

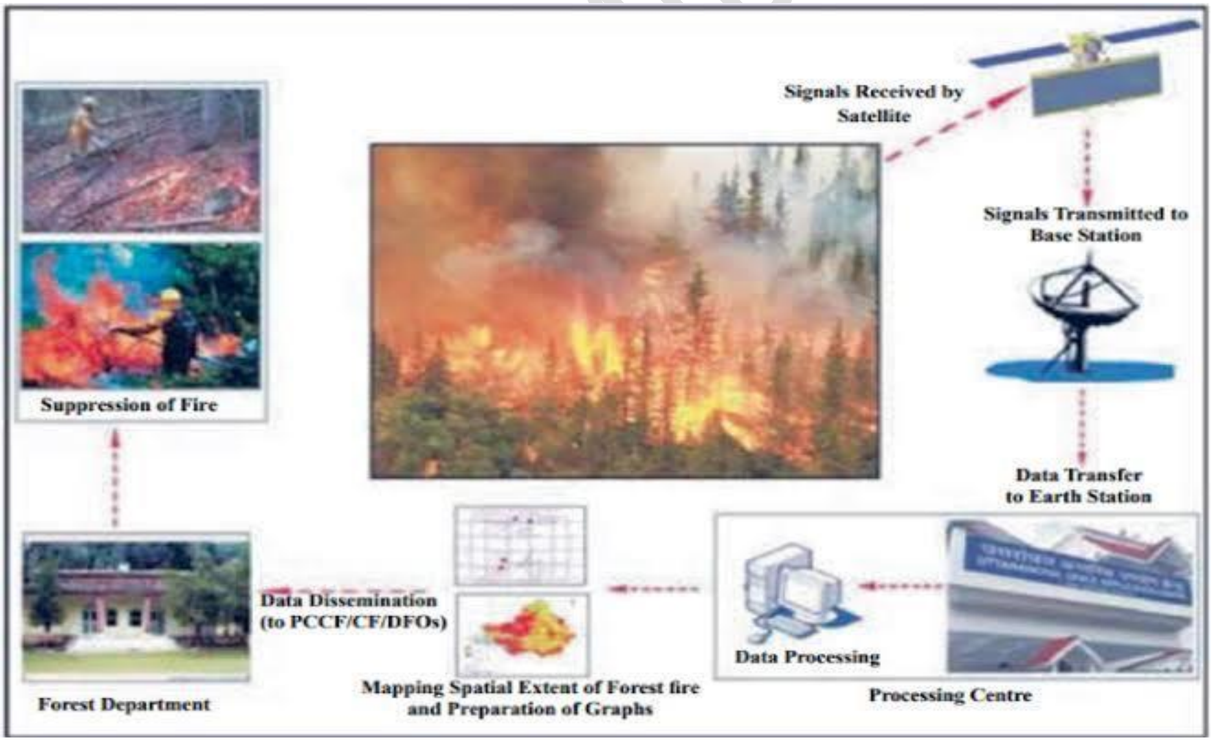
SOLUTION ARCHITECTURE:

Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

• Find the best tech solution to solve existing business problems and describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.

• Define features, development phases, and solution requirements and provide specifications according to which the solution is defined, managed and delivered.

TECHNICAL ARCHITECTURE:



5.3 USER STORIES

DEVELOPMENT WORK:

- **Data collection:** It is essential thing to gather the accuracy of the data or else the prediction can become incorrect. One of the important works in this is to collect the data of the forest like smoke, temperature, rain, dry season and wet season. It has high priority. It comes under Sprint 1.

- **Application of algorithm:** To gather the algorithms and determine each algorithm's accuracy level. Find potential prediction-based algorithms. It has high priority. It comes under Sprint 2.
- **Accuracy of the algorithm:** The accuracy of each algorithm is calculated and making it simple to get the most precise results Then we need to check for the accuracy of those algorithms. It has a medium priority. It comes under Sprint 2. The dataset that is collected in the first has been taken here for evaluation. Data has been accessed first before processing it. It has a medium priority. It comes under Sprint 1.
- **Determine the accuracy of the algorithm:** Determine the accuracy, precision and recollection of each algorithm. Output has been predicted using it. It has high priority. It comes under Sprint 3.

ECO ACTIVITIES:

- **Core purpose:** All devices and screen sizes can access the website without any issues. The program should be designed and developed in such a way that the optimal user interface and maintenance are considered. The updates must contain the answers to the asked questions in a timely manner. It has high priority. It comes under Sprint 1.

CHAPTER-6

PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

SPRINT PLANNING:

SPRINT-1

- **Data collection:** One of the important works in this is to collect the data of the forest like smoke, temperature, rain, dry season and wet season. It has 2 story points and has high priority.
- **Application of algorithm:** Find potential prediction-based algorithms. It has 2 story points and has high priority.

SPRINT-2

- **Accuracy of the algorithm:** Then we need to check for the accuracy of those algorithms. It has 1 story point and has medium the priority.
- **Determine the accuracy of the algorithm:** Determine the accuracy, precision and recollection of each algorithm. It has 1 story points and has medium priority.

SPRINT-3

- **Core purpose:** The program should be designed and developed in such a way that the optimal user interface and maintenance are considered. It has 2 story points and has medium priority. All devices and screen sizes can access the website without any issues. It has 1 story point and has low priority. The updates must contain the answers to the asked questions in a timely manner. It has 1 story point and has low priority.

SPRINT ESTIMATION:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

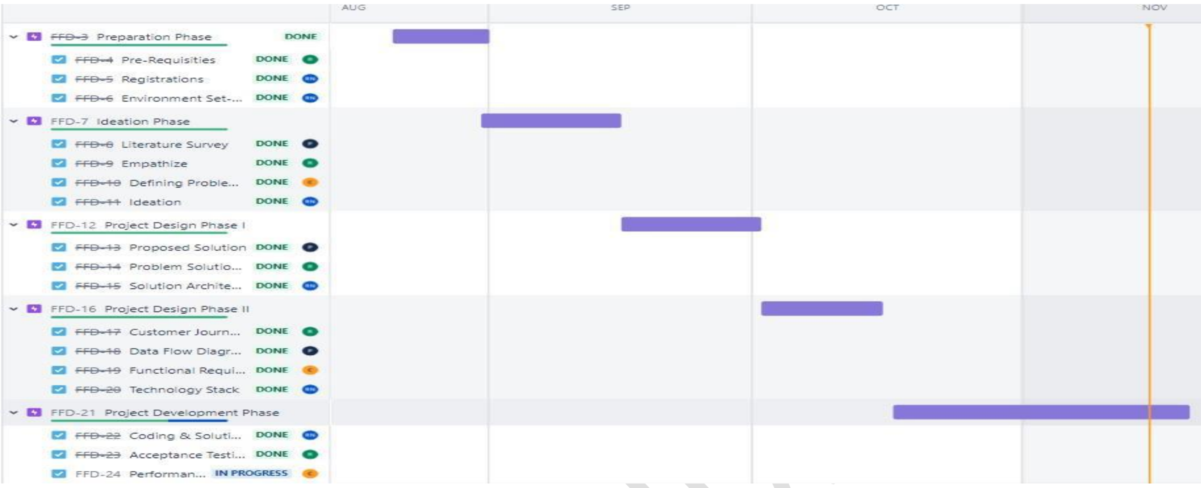
$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

= 60/10=6

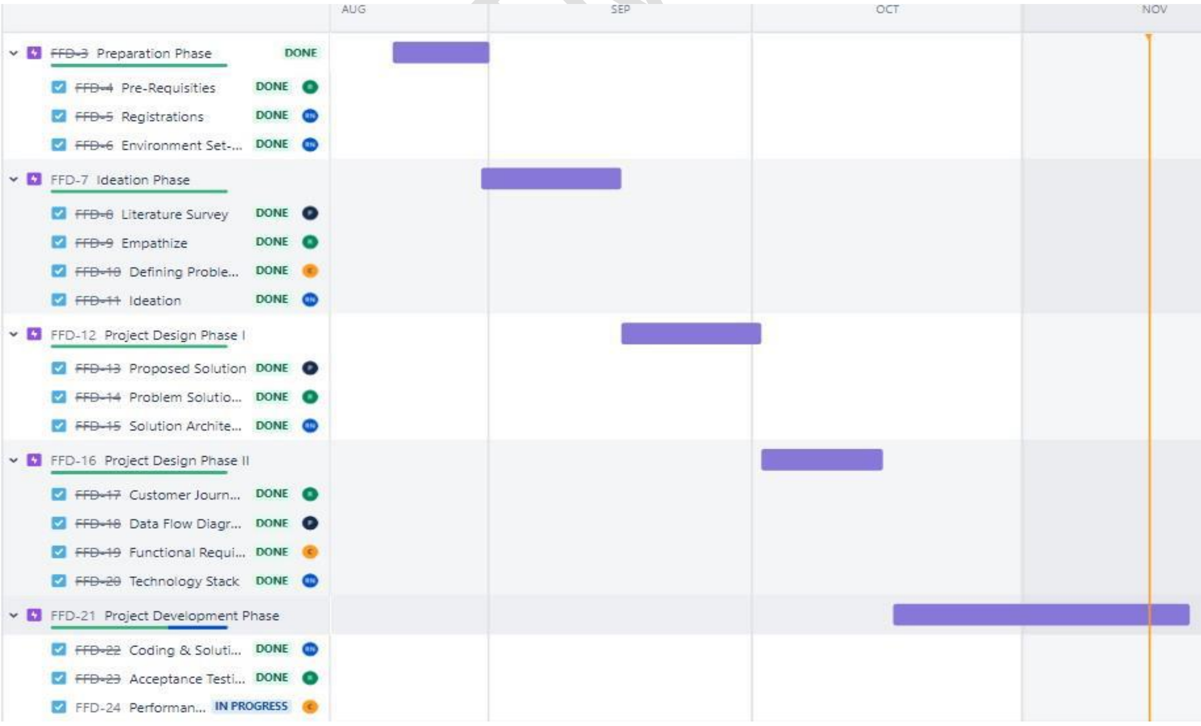
AV = 6

6.2 SPRINT DELIVERY SCHEDULE:

Sprint planning is an event in scrum that kicks off the sprint. The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved. Sprint planning is done in collaboration with the whole scrum team.



6.3 REPORT FROM JIRA



CHAPTER-7

CODING & SOLUTIONING

7.1 CODING

```
import tensorflow
import keras
from keras.preprocessing.image import ImageDataGenerator
train_datagen=ImageDataGenerator(rescale=1./255,shear_range=0.2,rotation_ra
nge=180,zoom_range=0.2, horizontal_flip=True)
test_datagen=ImageDataGenerator(rescale=1./255)
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It
includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='Hu0crJth4iJIgd922IJK46d06bVFaEwYc-4rmxAF7-sm',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-
storage.appdomain.cloud')

bucket = 'emergingmethodsforforestfiredetec-donotdelete-pr-meznojcru6qftr'
object_key = 'train_set.zip'

streaming_body_8 = cos_client.get_object(Bucket=bucket,
Key=object_key)['Body']

import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='Hu0crJth4iJIgd922IJK46d06bVFaEwYc-4rmxAF7-sm',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
```

```
    endpoint_url='https://s3.private.us.cloud-object-
storage.appdomain.cloud')

bucket = 'emergingmethodsforforestfiredetec-donotdelete-pr-meznojcru6qftr'
object_key = 'test_set.zip'

streaming_body_9 = cos_client.get_object(Bucket=bucket,
Key=object_key)['Body']

from io import BytesIO
import zipfile
train_unzip =zipfile.ZipFile(BytesIO(streaming_body_8.read()),'r')
train_file_paths=train_unzip.namelist()
for path in train_file_paths:
    train_unzip.extract(path)

from io import BytesIO
import zipfile
test_unzip =zipfile.ZipFile(BytesIO(streaming_body_9.read()),'r')
test_file_paths=test_unzip.namelist()
for path in test_file_paths:
    test_unzip.extract(path)

import os
file_path=os.listdir('/home/wsuser/work/train_set')

import os
test_file_path=os.listdir('/home/wsuser/work/test_set')

x_train=train_datagen.flow_from_directory('/home/wsuser/work/train_set',tar
get_size=(64,64),batch_size=32, class_mode='binary')

Output: Found 436 images belonging to 2 classes.

x_test=test_datagen.flow_from_directory('/home/wsuser/work/test_set',target
_size=(64,64),batch_size=32, class_mode='binary')

Output: Found 121 images belonging to 2 classes.

x_test.class_indices

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

#To define Linear initialisation import Sequential
from keras.models import Sequential
#To add layers import Dense
from keras.layers import Dense
#To create Convolution kernel import Convolution2D
from keras.layers import Convolution2D
#import Maxpooling layer
```

```
from keras.layers import MaxPooling2D
#import flatten layer
from keras.layers import Flatten

import warnings
warnings.filterwarnings('ignore')

model=Sequential()

model.add(Convolution2D(32,(3,3),input_shape=(64,64,3),activation='relu'))
#add maxpooling layer
model.add(MaxPooling2D(pool_size=(2,2)))
#add flatten layer
model.add(Flatten())
#add hidden layer
model.add(Dense(150,activation='relu'))
#add output layer
model.add(Dense(1,activation='sigmoid'))
model.compile(loss='binary_crossentropy',optimizer="adam",metrics=["accurac
y"])

model.fit_generator(x_train,steps_per_epoch=14,epochs=5,validation_data=x_t
est,validation_steps=20)
```

```
Output: Epoch 1/5 14/14 [==============================] - ETA: 0s - loss:
0.7870 - accuracy: 0.6858WARNING:tensorflow:Your input ran out of data;
interrupting training. Make sure that your dataset or generator can
generate at least `steps_per_epoch * epochs` batches (in this case, 20
batches). You may need to use the repeat() function when building your
dataset. 14/14 [==============================] - 17s 1s/step - loss:
0.7870 - accuracy: 0.6858 - val_loss: 0.1734 - val_accuracy: 0.9504 Epoch
2/5 14/14 [==============================] - 11s 837ms/step - loss: 0.2771
- accuracy: 0.8807 Epoch 3/5 14/14 [==============================] - 12s
844ms/step - loss: 0.2318 - accuracy: 0.9083 Epoch 4/5 14/14
[==============================] - 12s 827ms/step - loss: 0.1874 -
accuracy: 0.9289Epoch 5/5 14/14 [==============================] - 12s
832ms/step - loss: 0.1728 - accuracy: 0.9197

<keras.callbacks.History at 0x7f3150a41ac0>
```

```
model.save("/home/wsuser/work/forest1.h5")

#import load_model from keras.model
from keras.models import load_model
#import image class from keras
from tensorflow.keras.preprocessing import image #import numpy
import numpy as np
#import cv2
import cv2
```

```
pred= model.predict(x_test)
pred=np.round(pred)

pred
```

Output: array([[1.], [0.], [1.], [0.], [0.], [0.], [0.], [1.], [0.], [1.], [0.], [0.], [0.], [0.], [0.], [1.], [1.], [0.], [0.], [1.], [1.], [0.], [0.], [1.], [0.], [1.], [0.], [0.], [0.]], dtype=float32)

```
print(len(pred))
```

Output: 121

```
#import load_model from keras.model
from keras.models import load_model
#import image class from keras
import tensorflow as tf
from tensorflow.keras.preprocessing import image
#import numpy
import numpy as np
#import cv2

model = load_model("/home/wsuser/work/forest1.h5")

def predictImage(filename):
  img1 = image.load_img(filename,target_size=(64,64))
  Y = image.img_to_array(img1)
  X = np.expand_dims(Y,axis=0)
  val = model.predict(X)
  print(val)
  if val == 1:
    print(" fire")
  elif val == 0:
      print("no fire")

predictImage("/home/wsuser/work/test_set/with fire/19464620_401.jpg")
```

Output: [[1.]] fire

```
#import opencv librariy
#import cv2
#import numpy
import numpy as np
#import image function from keras
from keras.preprocessing import image
#import load_model from keras
from keras.models import load_model
#import client from twilio API
```

```
from twilio.rest import Client
#imort playsound package
from playsound import playsound
#load the saved model
model = load_model(r'/home/wsuser/work/forest1.h5')
#define the features

name = ['forest','with forest']
account_sid='AC7fbd9e1b65a166f13459d8eca7b664cf'
auth_token='8e7e8e6672a8fb0a908ab3137560022d'
client=Client(account_sid,auth_token)
message=client.messages \
.create(
 body='Forest Fire is detected, stay alert',
 from_='+18434385489',
 to='+91 95666 05556'
)
print(message.sid)
```

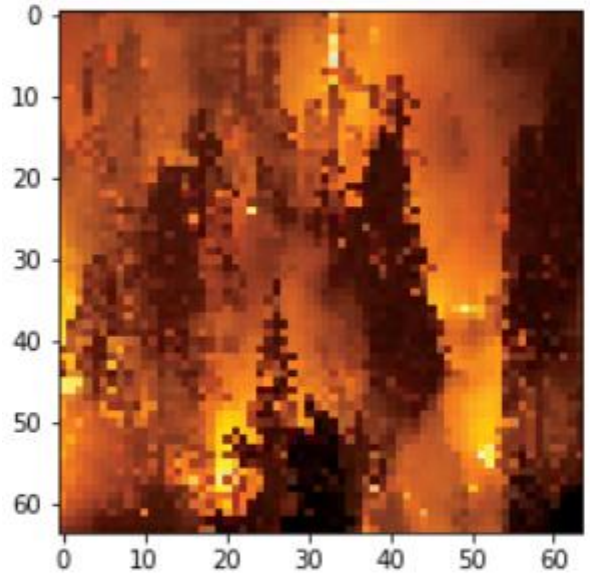
Output: SM2da9ccdbd712f3d9ff08e82c93f1411f

```
def message(val):
  if val==1:
    from twilio.rest import Client
    print('Forest fire')
    account_sid='AC7fbd9e1b65a166f13459d8eca7b664cf'
    auth_token='8e7e8e6672a8fb0a908ab3137560022d'
    client=Client(account_sid,auth_token)
    message=client.messages \
     .create(
        body='forest fire is detected, stay alert',
        #use twilio free number
        from_='+18434385489',
        #to number
        to='+91 95666 05556')
    print(message.sid)
    print("Fire detected")
    print("SMS Sent!")
  elif val==0:
    print('No Fire')

from matplotlib import pyplot as plt
#import load model from keras.model
from keras.models import load_model
#import image from keras
from tensorflow.keras.preprocessing import image
img1 = image.load_img('/home/wsuser/work/test_set/with
fire/Wild_fires.jpg',target_size=(64,64))
Y = image.img_to_array(img1)
```

```
x = np.expand_dims(Y,axis=0)
val = model.predict(x)
plt.imshow(img1)
plt.show()
message(val)
```

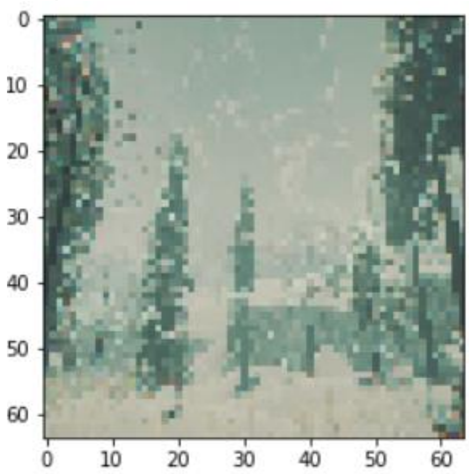
Output:



Forest fire SMa59dccec808fcfc4feecf7c556209214 Fire detected SMS Sent!

```
img2 =
image.load_img('/home/wsuser/work/test_set/forest/1200px_Mountainarea.jpg',
target_size=(64,64))
Y = image.img_to_array(img2)
x = np.expand_dims(Y,axis=0)
val = model.predict(x)
plt.imshow(img2)
plt.show()
message(val)
```

Output:



No Fire

```
from ibm_watson_machine_learning import APIClient
wml_credentials={"url":"https://us-
south.ml.cloud.ibm.com","apikey":"JwrUkG_NWgWEoonXcz4EIJSJzXWbH97kolVXGvjt9
Apr"}
client=APIClient(wml_credentials)
def guid_from_space_name(client,space_name):
    space=client.spaces.get_details()
    return(next(item for item in space['resources']if
item['entity']["name"]==space_name)['metadata']['id'])
space_uid=guid_from_space_name(client,'project')
print("Space UID= "+space_uid)
```

Output: Space UID= e8fea989-74c5-4b46-8f86-4fe706dd6e22

```
client.set.default_space(space_uid)
```

Output: 'SUCCESS'

```
client.software_specifications.list()
```

Output: NAME ASSET_ID TYPE default_py3.6 0062b8c9-8b7d-44a0-a9b9-46c416adcbd9 base kernel-spark3.2-scala2.12 020d69ce-7ac1-5e68-ac1a-31189867356a base pytorch-onnx_1.3-py3.7-edt 069ea134-3346-5748-b513-49120e15d288 base scikit-learn_0.20-py3.6 09c5a1d0-9c1e-4473-a344-eb7b665ff687 base spark-mllib_3.0-scala_2.12 09f4cff0-90a7-5899-b9ed-1ef348aebdee base pytorch-onnx_rt22.1-py3.9 0b848dd4-e681-5599-be41-b5f6fccc6471 base ai-function_0.1-py3.6 0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda base shiny-r3.6 0e6e79df-875e-4f24-8ae9-62dcc2148306 base tensorflow_2.4-py3.7-horovod 1092590a-307d-563d-9b62-4eb7d64b3f22 base

```
software_spec_uid=client.software_specifications.get_uid_by_name("tensorflo
w_1.15-py3.6")
```

```
software_spec_uid
```

Output: '2b73a275-7cbf-420b-a912-eae7f436e0bc'

```
Keras
```

Output: <module 'keras' from '/opt/conda/envs/Python-3.9/lib/python3.9/site-packages/keras/__init__.py'>

```
!tar -zcvf forest-fire-detection-model_new.tgz forest1.h5
```

Output: forest1.h5

```
ls -1
```

Output: forest1.h5 forest-fire-detection-model_new.tgz **test_set**/ **train_set**/

CHAPTER-8

TESTING

8.1 TEST CASES

TEST_CASE_ID-1:

- **FEATURE TYPE**: Functional
- **TEST SCENARIO 1:** Satellites gather photographs and relay them to the monitoring authority based on the images. It gathers the images with fire.
- **Expected result**: Forest images are gathered. Actual result is working as expected and the status is pass.
- **TEST SCENARIO 2:** satellite remote sensing used to detect fire. During a fire event, active fires can be detected by detecting the heat, light and smoke plumes emitted from the fires.
- **Expected result:** It detects the fire using the heat,light and smoke that are emitted while the fire is started spreading. Actual result is working as expected and the status is pass.
- **TEST SCENARIO 3:** For this problem we use image processing and video analysis so by using satellite image processing. we can able to find the fire at the early stage and stop spreading fire in the forest.
- **Expected result:** Early stage for detecting it using the symptoms is easy to safe more places from forest fire. Actual result is working as expected and the status is pass.
- **TEST SCENARIO 4:** Human-caused fires result from campfires left unattended, the burning of debris and intentional acts of arson. This model is mainly build by using CNN and machine learning and deep learning .
- **Expected result:** It plays a vital role using these technique and utilized in a efficient way to protect the forest from fire. Actual result is working as expected and the status is pass.

TEST_CASE_ID-2:

- **FEATURE TYPE:** UI
- **TEST SCENARIO:** Sensors are used to monitor the nature of air. Sensor detects the place where fire occurs in the forest.
- **Expected result:** Detects the places that are actively filled with fire. Actual result is working as expected and the status is pass.
- **Bug id:** BUG-1234

TEST SCENARIOS

- Verify user is able to see login page
- Verify user is able to loginto application or not?
- Verify user is able to navigate to create your account page?
- Verify user is able to recovery password
- Verify login page elements

SEARCH

- Verify user is able to search by entering keywords in search box
- Verify user is able to see suggestions based on keyword entered in search box
- Verify user is able to see related auto suggestions displaying based on keyword entered in search box
- Verify user is able to see no matches found message when no results are matching with entered keyword
- Verify user is able to see seach detailed page when nothing entered in textbox

8.1 USER ACCEPTANCE TESTING

- **PURPOSE OF DOCUMENT:** The purpose of this document is to briefly explain the test coverage and open issues of the [EMERGING METHOD FOR EARLY FOREST FIRE DETECTION] project at the time of the release to User Acceptance Testing (UAT).
- **DEFECT ANALYSIS:** This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	3	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	10	2	4	20	36
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	4	2	1	7
Totals	23	13	14	26	76

- **TEST CASE ANALYSIS:** This report shows the number of test cases that have passed, failed, and untested.

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	8	0	0	8
Client Application	63	0	0	63
Security	5	0	0	5
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

CHAPTER-10

ADVANTAGES

- Fire detection systems increase response times, as they are able to alert the correct people in order to extinguish the fire.
- This thus reduces the amount of damage to the property. Fire detection systems can be connected to sprinklers that will automatically respond when a fire is detected.
- This project can automate the water when the fire is detected so it can prevent major area to get affects.
- Fire alarms are important because they can give you an early signal to something that could be tragic – basically saving your lives. A fire alarm alerts you when they you are busy, working or sleeping.

CHAPTER-11

CONCLUSION

Nowadays, forest fires became one of the foremost important problems that cause damage to several areas around the world. The paper displays machine learning regression techniques for predicting forest fire-prone areas. This reduces the chances of false alarm to a great extent. Our system is able to differentiate various forest fire scenarios, from initial case (no fire) to detection of fire, fairly accurately. It can accurately determine the growth of fire. This will help in early stages of fire detection and help to confine fire to limited areas before much damage occurs. The system will be very effective in preventing occurrence of false alarms. We aim at monitoring the forests without constant human supervision.

CHAPTER-11

FUTURE SCOPE

In this paper, machine learning Infrastructure for forest fire detection system has been designed to detect fire at earlier stage.The most important objective in fire surveillance is early and reliable detection and localization of the fire. As a result the proposed model is tested with sensor data is recorded for testing of fire detection. The system will process the information and send mail using GSM module to the nearby fire service station with the location of the fire using GPS module. Early detection and prediction will lower the count of forest fires in the entire world and save our planet Earth. In future, we can develop this model to minimize the energy consumption of all sensors and complete networks considering the node distribution among clusters using distributed sensing.