**PROJECT REPORT**

**EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRE**

TEAM ID: **PNT2022TMID04190**

***submitted by***

| **KARTHIK SRI SAKTHI DEGALA** | **412519205066** |
| --- | --- |
| **PRIYA DHARSHINI R** | **412519025105** |
| **JEEVAPRAKASH T** | **412519205056** |
| **SRINITHI SAMYUKTHA S J** | **412519205138** |

**TABLE OF CONTENTS**

**1INTRODUCTION 1**

* 1. [PROJECT OVERVIEW 1](#_heading=h.gjdgxs)
  2. [PURPOSE 1](#_heading=h.30j0zll)

1. **LITERATURE SURVEY 2**
   1. [EXISTING PROBLEM 2](#_heading=h.1fob9te)
   2. [REFERENCES](#_heading=h.3znysh7) 5
   3. [PROBLEM STATEMENT DEFINITION](#_heading=h.2et92p0) 5
2. **IDEATION AND PROPOSED SOLUTION 7**
   1. [EMPATHY MAP CANVAS](#_heading=h.tyjcwt) 7
   2. [IDEATION & BRAINSTORMING](#_heading=h.3dy6vkm) 8
   3. PROPOSED SOLUTION 9
   4. PROBLEM SOLUTION FIT 10
3. **REQUIREMENT ANALYSIS 11**
   1. FUNCTIONAL REQUIREMENTS 11
   2. [NON FUNCTIONAL REQUIREMENTS 1](#_heading=h.1t3h5sf)2
4. **PROJECT DESIGN 13**
   1. [DATA FLOW DIAGRAM 1](#_heading=h.4d34og8)3
   2. [SOLUTION & TECHNICAL ARCHITECTURE 1](#_heading=h.2s8eyo1)4
   3. [USER STORIES 1](#_heading=h.17dp8vu)6
5. **PROJECT PLANNING AND SCHEDULING 17**
   1. [SPRINT PLANNING AND ESTIMATION 1](#_heading=h.3rdcrjn)7
   2. [SPRINT DELIVERY SCHEDULE 1](#_heading=h.26in1rg)9
6. **CODING & SOLUTIONING 20**
7. **TESTING 27**
   1. [TEST CASES 2](#_heading=h.lnxbz9)7
   2. [USER ACCEPTANCE TESTING 2](#_heading=h.35nkun2)8

1. **RESULTS 29**
   1. [PERFORMANCE METRICS 2](#_heading=h.1ksv4uv)9
2. **ADVANTAGES & DISADVANTAGES 30**

[ADVANTAGES](#_heading=h.44sinio)

[DISADVANTAGES](#_heading=h.2jxsxqh)

1. **CONCLUSION 31**

**12FUTURE SCOPE 32**

[**APPENDIX 3**](#_heading=h.z337ya)**3**

[SOURCE CODE 3](#_heading=h.3j2qqm3)3

[GITHUB](#_heading=h.1y810tw) 36

PROJECT DEMO 36

# CHAPTER 1

## INTRODUCTION

### PROJECT OVERVIEW

Fire can make major hazards in this hectic world. All buildings and vehicles used in public transportation have fire prevention and fire protection systems due to the accelerated number in the fire incidents. Also, many of the firms conduct a mock fire drill in every occurrence of months to protect their employees from the fire. This would help them to understand what to do or what not to do when a fire situation happens. Forests are one of the main factors in balancing the ecology. It is very harmful when a fire occurs in a forest. But most of the time, the detection of forest fire happens when it spread over a wide region. Sometimes, it could not be possible to stop the fire. As a result, the damage of the environment is higher than predictable. The emission of large amount of carbon dioxide (CO2) from the forest fire damages the environment. As well as it would lead to complete disappearance of rare species in the world. Also, it can make an impact on the weather, and this make major issues like earthquakes, heavy rains, floods and so on.A research study shows an automatic fire detection can be divided into three groups: aerial, ground and borne detection. The ground-based systems use several staring black and white video cameras are used in fire detection which detect the smoke and compares it with the natural smoke. The main benefit of using this system is high temporal resolution and spatial resolution. So that, the detection is easier.2But these mechanisms still have some drawbacks in detecting the early stage of the fire. So that, it is highly important to introduce a system to detect the fire early as possible.Moreover, information regarding the seat of the hearth is invaluable for the rapid deployment of ﬁre- ﬁghters. Therefore, early detection, containment at the primary stages and extinguishment of a ﬁreplace before it spreads are crucial for wildﬁre Management.

.

### PURPOSE

Forest ﬁres as of late have been annihilating both for normal biological system, biodiversity and woodland economy. With expanding populace weight and change in worldwide atmosphere situation, there is an expansion in level of ﬁres that are a signiﬁcant reason for declining Indian woodlands. As indicated by woodland study report of India, 50 % of backwoods regions in nation are ﬁre inclined (going from 50 to 90 % in certain conditions of nation). Around 6 % of the woods are inclined to extreme ﬁre harms.

# CHAPTER 2

## LITERATURE SURVEY

### EXISTING PROBLEM

The existing system for detecting ﬁre are smoke alarms and heat alarms. The main disadvantage of the smoke sensor alarm and heat sensor alarms are that just one module is not enough to monitor all the potential ﬁre prone places. The only way to prevent ﬁre is too cautious at the time. Even if they are installed in every nook and corner, it just is not suﬃcient for an eﬃcient output consistently. As the number of smoke sensor requirement increase the cost will also increase to its multiple. The proposed system can produce consistent and highly accurate alerts within seconds of accident of the ﬁre. It reduces cost because only one software is enough to power the entire network of surveillance. Research is active on this ﬁeld by data scientists and machine learning researchers. The real challenge is to minimize the error in detection of ﬁre and sending alerts at the right time. The idea of this research is to fabricate a system through IoT sensors, which is arbitrarily spread in the forest and to make a self-sorted out powerful system between the sensors to cover all the enormous

territories in the forest that will used to maintain a strategic distance from the ﬁre harm whenever. The capacity of the sensor is to identify ﬁre in the inclusion region between the time intermission of each 5-10 minutes. At the point when the ﬁre is recognized the entirety of the sensor in the region will be dynamic and order to stop the normal assignment. The concept is to build early ﬁre detector using Arduino which is connected with different IoT sensors. Putting all efforts to develop a smarter system by connecting it to a webpage and monitoring the developed system statistics controlled by the Arduino programming. The use of latest technology can help to prevent the catastrophic accidents in forests. The aim is to early detect the ﬁreplace in forest by considering the several factor like smoke, temperature, humidity, ﬂame and based on the data we get from this programming, the forest department will be able to take an appropriate decision and the rescue team will be able to arrive on time at exact location. Consider, if it is a large region and it produces more carbon monoxide than the ordinary vehicle traﬃc. Surveillance of the danger areas and an early detection of ﬁreplace can appreciably shorten the response time and additionally decrease the practicable injury as nicely as the fee of ﬁreﬁghting. Known rule applies here: 1 minute – 1 cup of water, 2 minutes - 100 liters of water, 10 minutes - 1000 liters of water. The goal is to notice the ﬁreplace as quicker as possible, its actual localization and early notiﬁcation to the ﬁre devices. When ﬁre starts then the ﬂammable texture may likewise issues fuel to the hearth focal spot. The spot at that point will expand and more extensive. The ﬁrst phase

of start is alluded as "surface ﬁre‟ stage. This may feed on abutting bushes and the ﬁre will turn into higher and transforming into "crown ﬁre‟. Generally, at this stage the hearth transforms into wild and injury which end up being extreme that could stay for quite long time while depending on atmosphere conditions and the territory. Forest ﬁre detection using optimized solar–powered ZigBee wireless sensor networks- In this paper, they have developed system for Forest Fire Detection which overcomes the demerits of the Existing technologies of Forest Fire Detection. It can be ensured that the system developed can be implemented on a large scale with its promising results. The system is provided with low-power elements, higher versions of Zigbee, Maximum power point tracking Algorithm is used in order to make the system run for longer periods eﬃciently. Forest ﬁres are a very serious problem in many countries, and global warming may contribute to make this problem worse. Experts agree that, in order to prevent these tragedies from happening, it is necessary to invest in new technologies and equipment that enable a multifaceted approach. This paper describes a WSN for early detection of forest ﬁres. This network can be easily deployed at areas of special interest or risk. There are two types of nodes from the physical structure point of view: SNs, to collect data from the environment, and CNs, to gather data from the SNs and transmit the information to a Control Centre. The nodes also can be in different functioning modes. This enables a proper and seamless conﬁguration of the network, provides redundancy, and ensures there will be full temporal and geographical coverage in the deployment zone. The information gathered is related not only to early detection purposes but also to environment monitoring to maximize the WSN usage. This environmental data can also be employed to ﬁreﬁghting preventive tasks such as vegetation modelling, microclimate studies, and propagation model parametrization.

In this paper, a forest ﬁre detection algorithm is proposed. The algorithm uses YCbCr color space since it effectively separates luminance from chrominance and is able to separate high temperature ﬁre center pixels because the ﬁre at the high temperature center region is white. The ﬁnal results show that the proposed system has good detection rates and fewer false alarms, which are the main crucial problems of the most existing algorithms. The presences of ﬁre in video streams are indicated by semantic events. Most of the existing systems can only be used for the videos obtained from stationary cameras and videos obtained from the controlled lightening conditions.

These existing automatic ﬁre detection systems cannot be used for video streams obtained from mobile phones or any handheld devices. It was decried as a global tragedy. Lit by farmers, the ﬁres raged through villages, destroyed ecosystems and pumped climate-warming pollution into the atmosphere.

### Construction:

The sensors cover two in terminal with an electrolyte. The electrodes are classically fictional by arrangements a highly costly character on to the penetrable hydrophobic pia

mater. The at work(predicate) electrode gain both the electrolyte and the chillout information which has to be supervised regularly through a open dura mater. The electrolyte most commonly habit is a rock acrimonious the electrodes and shelter are for the most part in a moldable saddlecloth which restrain a gasoline vestibule concavity for the petrol and electrical brush.

### INTERNET OF THINGS:

The internet of things (IoT) can be determine as the mass of material devices, buildings, vehicles and many paragraph that are fixed with sensors, software, cobweb connectivity, actuators, and electronics that suffer these sight for amass and interchanging complaint. In usual Internet of Things (IoT) is a framework that afford animals, aim or community, the capability to emit over data to a netting that may not enjoin the

Christian-to-electronic computer (H2C) or the humane-tohuman (H2H) interaction and the unparalleled identifiers.

### DATA MANAGEMENT:

Data charge is an exact air in Internet of Things (IoT). The compass of the furnish data and the activities complex in thumbing of those notice come judicious, when examine a circle of end interrelated and statically dealing all style of instruction. An utilizable space came for wireless communications hew makers when M2M number has been emit, which is also the endow technology for Internet of Things (IoT). This technology hobble free row of applications.

Some of the most relevant concepts which enable us to understand the challenges and opportunities of data management are:

* Data Collection and Analysis
* Big Data
* Semantic Sensor Networking
* Virtual Sensors
* Complex Event Processing.

### CONCLUSION

In this project changeable sensory parameters algorithmic rule, a system has been improved which will reduce the error perception and updates the deficiency to the expert often through the IOT landing. D2D association conventionality an definite integral part which intercept IOT surrounding to designate, accomplish, and support a endurable ecosystem. The system thus intend is powerful to expose the mixture variations, daring gases and fire event through the sensors in an diligence and powerful to update the complaint to the style expert through the IOT fulfill secondhand MQTT policy. The improved system can be unfold for tenement appliances and in industries also. However, the system above is meant for a sincere opinion news only. As a tomorrow aggravation,

several-decision company through the IOT landing is study a object and the exploration is

being done to effectuate this enormous toil. It is trust that with the technological advancements profitable in instant age scenario, the above rehearse several-opinion correspondence will also be unfold in aqiqiy delay environments.

### REFERENCES

* + 1. J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” Future Gener. Comput. Syst., vol. 29, no. 7, pp. 1645–1660, Sep. 2013.
    2. J. Buckley, “From RFID to the Internet of Things pervasive networked systems,” Conference Centre Albert Borschette (CCAB), Brussels, Belgium, Mar. 2006. [Online]. Available: ftp://ftp.cordis.europa.eu/pub/ ist/docs/ka4/au\_conf670306\_buckley\_en.pdf
    3. D. Evans, “The Internet of things: How the next evolution of the Internet is changing everything,” Cisco IBSG, San Francisco, CA, USA, Apr. 2011. [Online]. Available: <http://www.cisco.com/web/about/ac79/> docs/innov/IoT\_IBSG\_0411FINAL.pdf
    4. The Zettabyte Era-Trends and Analysis. Cisco, May 2013. [Online]. Available: <http://www.cisco.com/en/US/solutions/collateral/ns341/ns52>5/ ns537/ns705/ns827/VNI\_Hyperconnectivity\_WP.html
    5. D. Lake, A. Rayes, and M. Morrow, “The Internet of Things,” Internet Protocol J., vol. 15, no. 3, pp. 10–19, Sep. 2012. [Online]. Available: <http://www.cisco.com/web/about/ac123/ac147/archived_issu>es/ipj\_15-3/ 153\_Internet.html

### PROBLEM STATEMENT DEFINITION

**\*** In earlier times fires were detected with the help of watching towers or using satellite images.

\*Satellites collect images and send it to the monitoring authority which will decide by seeing images that it is a fire or not.

\*But this approach was very slow as the fire may have spread in the large areas and caused so much damage before the rescue team came.

\*In the watching tower method, there was a man always standing on the tower who would monitor the area and inform if there was fire.

\*This method was also slow because before the man got to know about the fire it may have spread in the inner parts of forest, also it always requires a man who must be present there.

\*Since, we know that some areas, especially forest areas are large so it is practically impossible to put a man in every part of forest from where they can monitor the forest area.

\*So, both these approaches of watching towers and satellite images failed to detect fire as early as possible to reduce the damage done by fire Problems in fire detection:

\*There were mainly two problems in fire detection as discussed:

1. Judging criteria for the fire: Edge is set, on the off chance that the worth is more noteworthy than edge, it is a fire, else not.

So, this problem was removed by using machine learning techniques by many researchers.

1. Connection of nodes: Traditional systems used cables to connect alarm with the detectors.

\*Cable was mainly of copper. But copper wire may be costly or it can suffer from fault in the mid-way.

\*So, this problem was removed using wireless sensor networks.

\*So, with the advancement in technology researchers find an efficient method to detect forest fire with the help of Wireless Sensor Network.

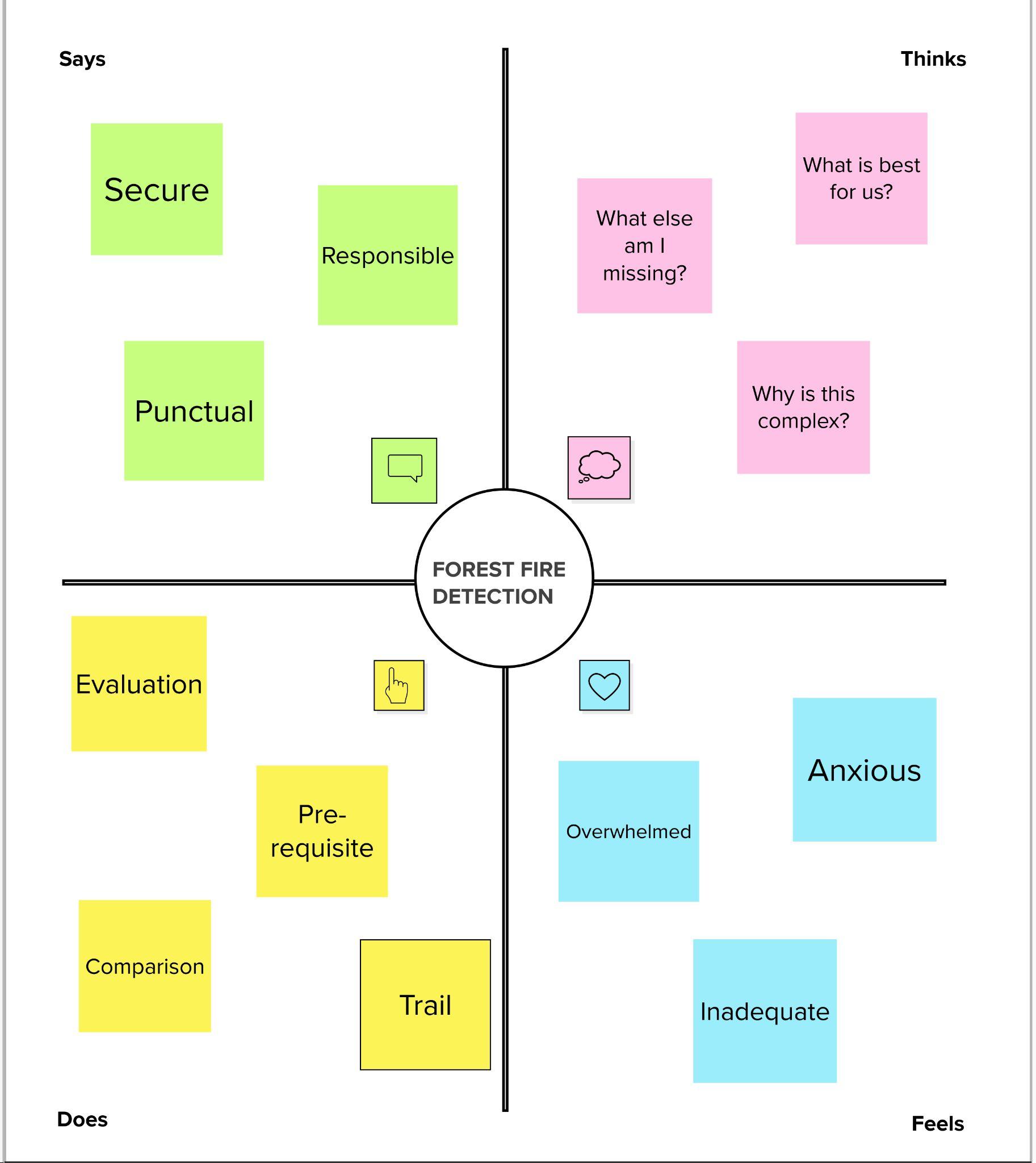
\*Fire can be identified by conveying sensor hubs in timberland regions by which they illuminate about fire.

Conveying sensor hubs in the timberland regions means placing sensors in every part of the forest and mostly in the prone areas where risk of 9 catching fire is more. With the use of wireless sensor networks, now it is easy to detect the fire in large areas as soon as possible.

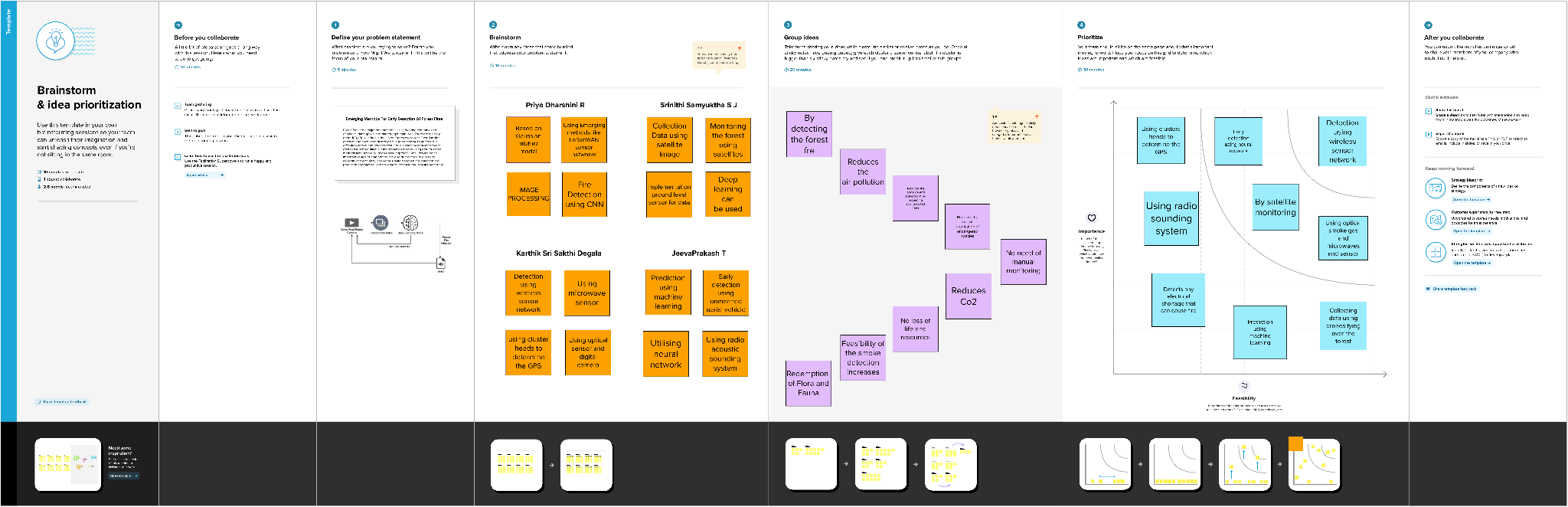
# CHAPTER 3

## IDEATION AND PROPOSED SOLUTION

### EMPATHY MAP CANVAS

****

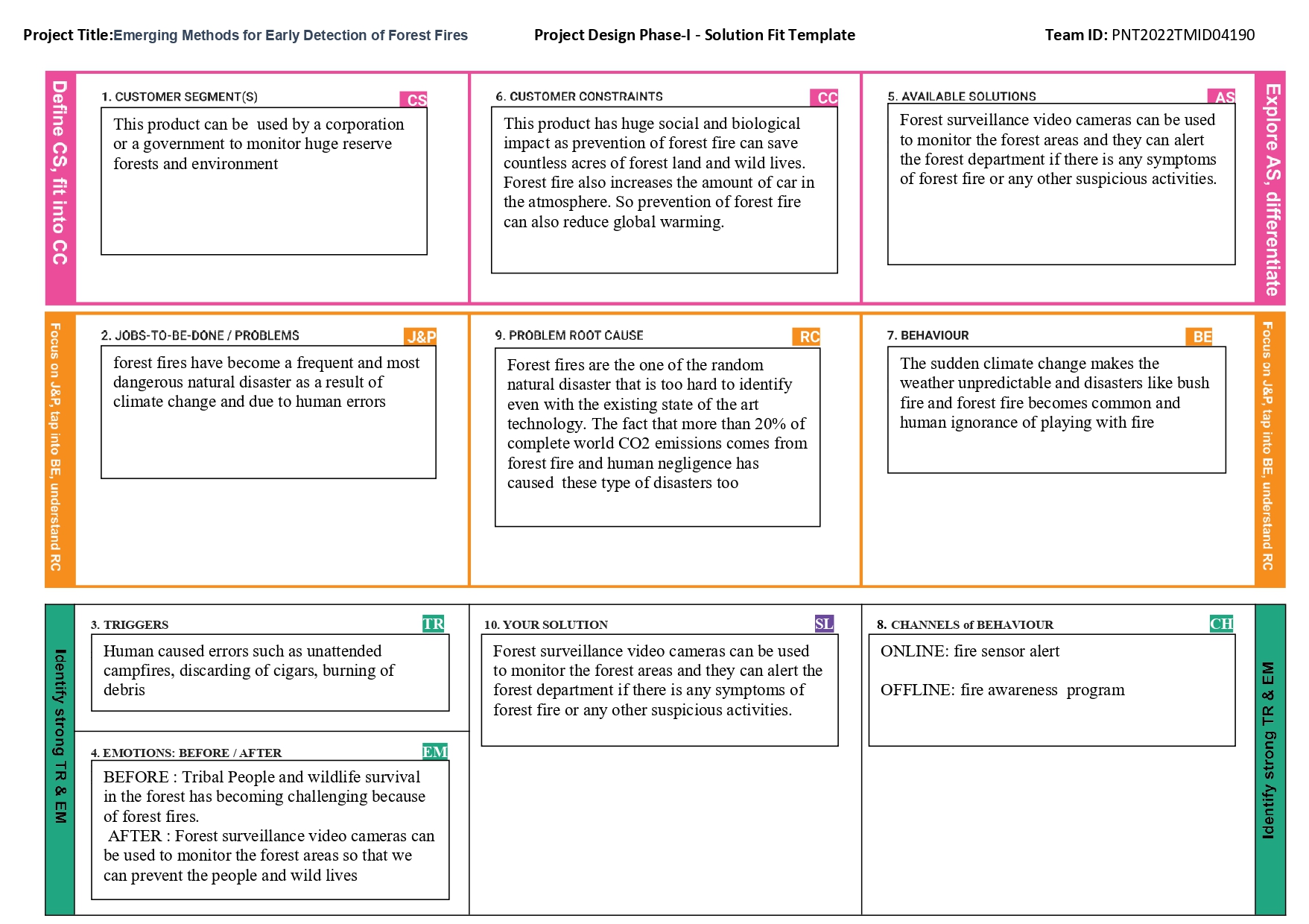
### IDEATION & BRAINSTORMING



### 3.3 Proposed Solution

| S.No  . | Parameter | Description |
| --- | --- | --- |
| 1. | 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. |
| 2. | 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. |
| 3. | Novelty / Uniqueness | Real time computer program detect forest fire in earliest before it spread to larger area. |
| 4. | Social Impact / Customer Satisfaction | Blocked roads and railway lines, electricity, mobile and land telephone lines cut, destruction of homes and industries. |
| 5. | Business Model (Revenue Model) | The proposed method was implemented using the Python programming language on a Core i3 or greater ( CPU and 4GB RAM.) |
| 6. | Scalability of the Solution | Computer vision models enable land cover classification and smoke detection from satellite and ground cameras. |

**3.4 PROBLEM SOLUTION FIT**

****

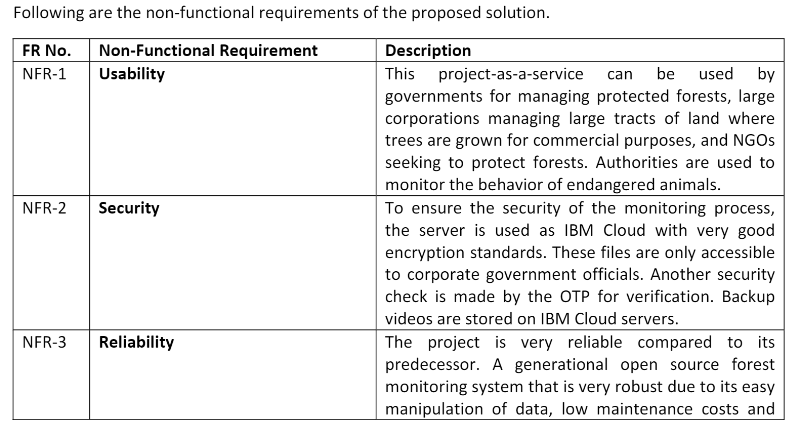
# CHAPTER 4

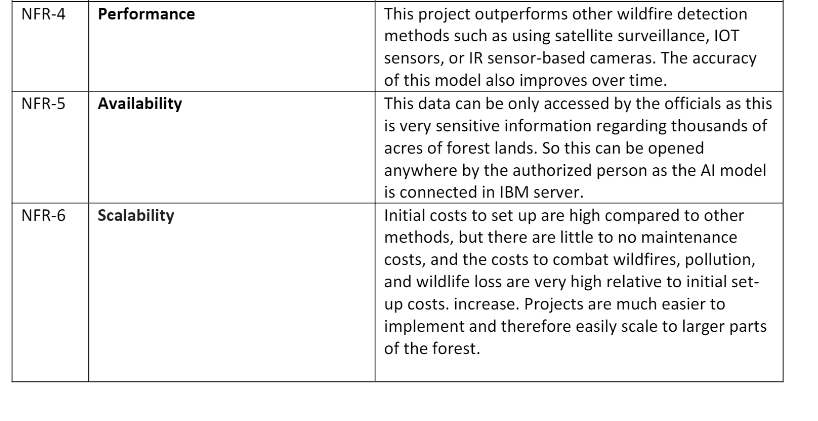
## REQUIREMENT ANALYSIS

### 4.1. FUNCTIONAL REQUIREMENT:

| FR  No. | Functional Requirement (Epic) | Sub Requirement (Story / Sub-Task) |
| --- | --- | --- |
| FR-1 | User Registration | Registration through Form  Registration through  Gmail |
| FR-2 | User Confirmation | Confirmation via Email  Confirmation via OTP |
| FR-3 | Overall Surveillance Report | Helps to understand the current scenario in the forest by giving report as “no fire” or “negative” |
| FR-4 | Cloud Server Access | To save and run the model from the camera footage |
| FR-5 | 5 Live Camera Feed | Real-time monitoring by the forest authorities |
| FR-6 | GSM Module | Warn the nearest forestry manager and local residents fire station |

### 4.2 Non-Functional requirement

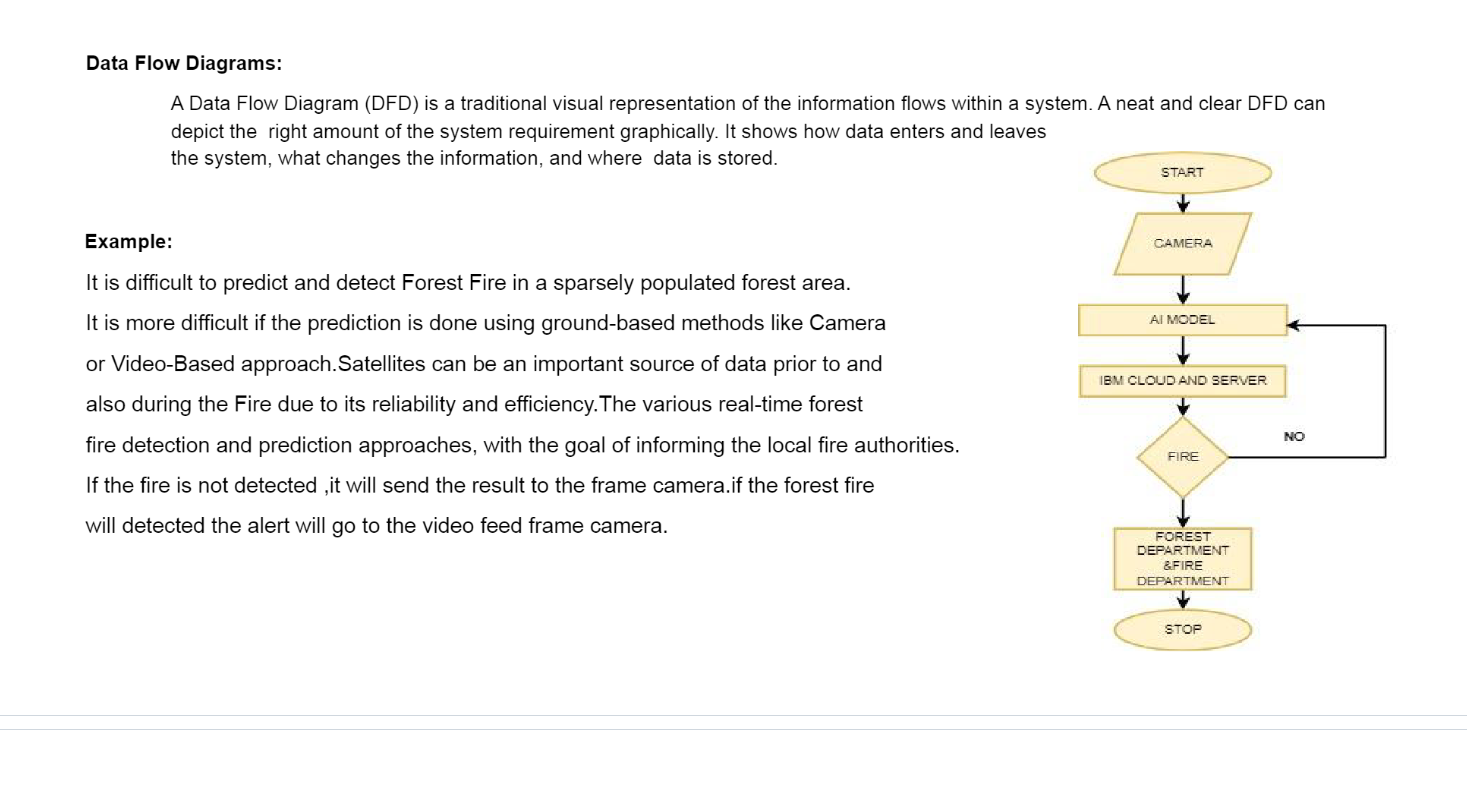




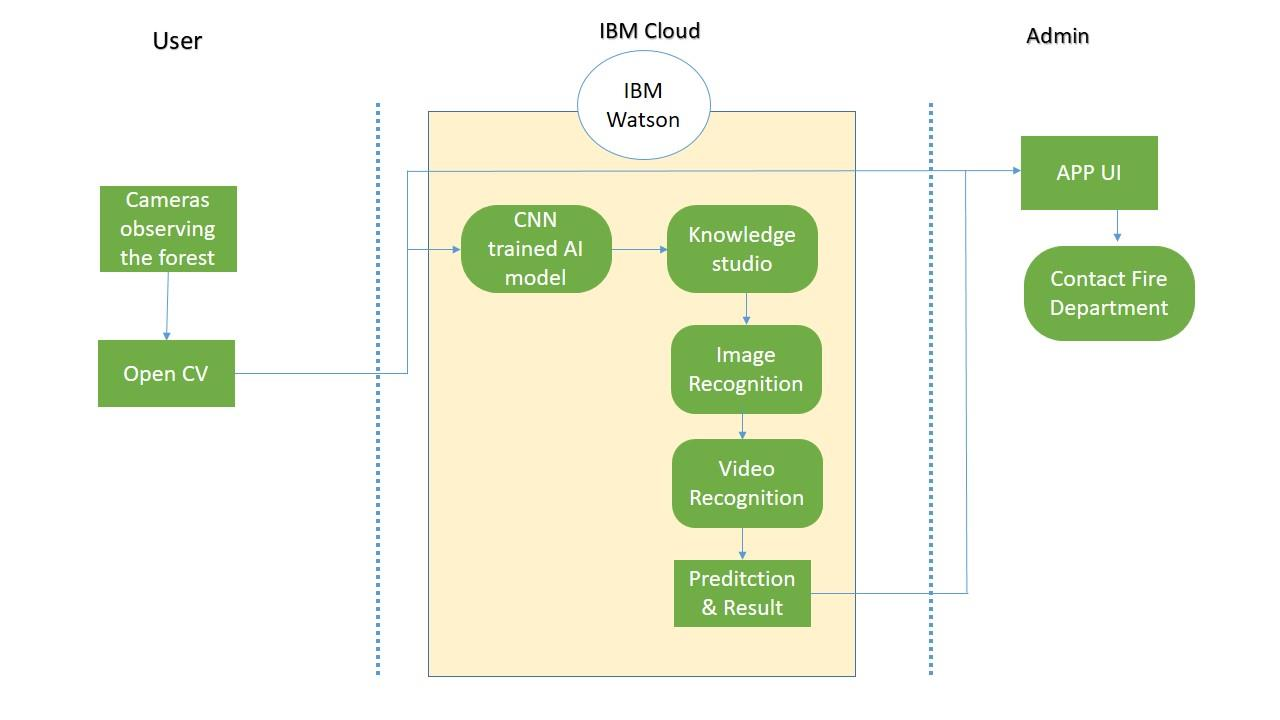
## CHAPTER 5

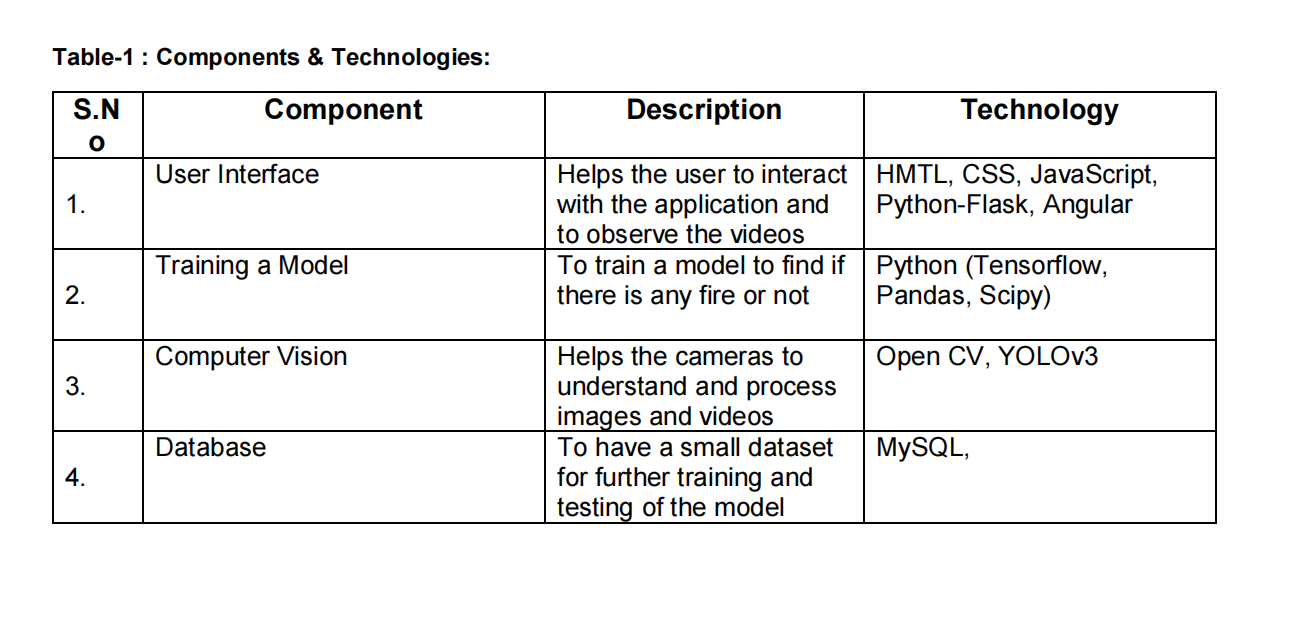
## PROJECT DESIGN

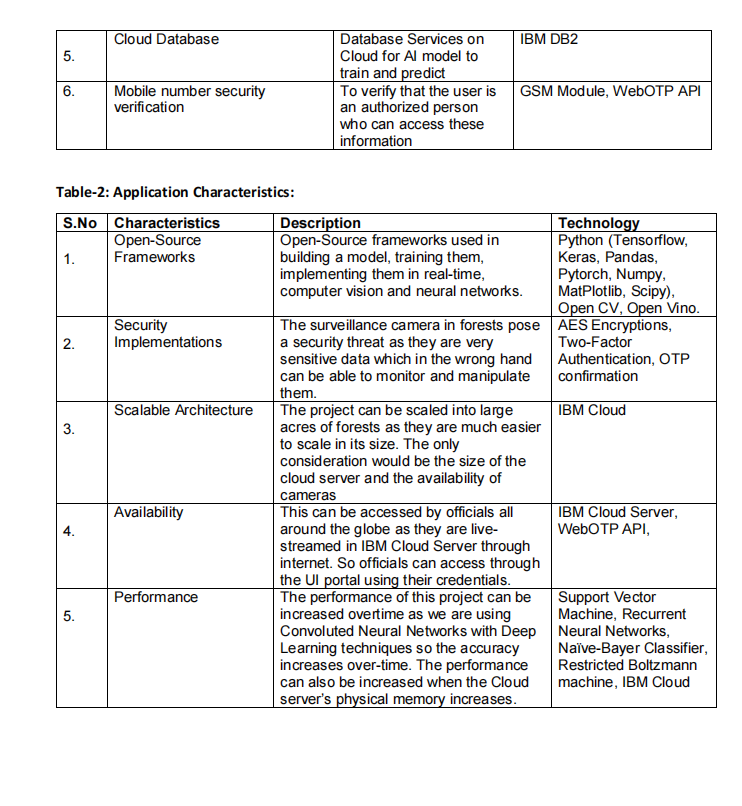
### DATA FLOW DIAGRAM



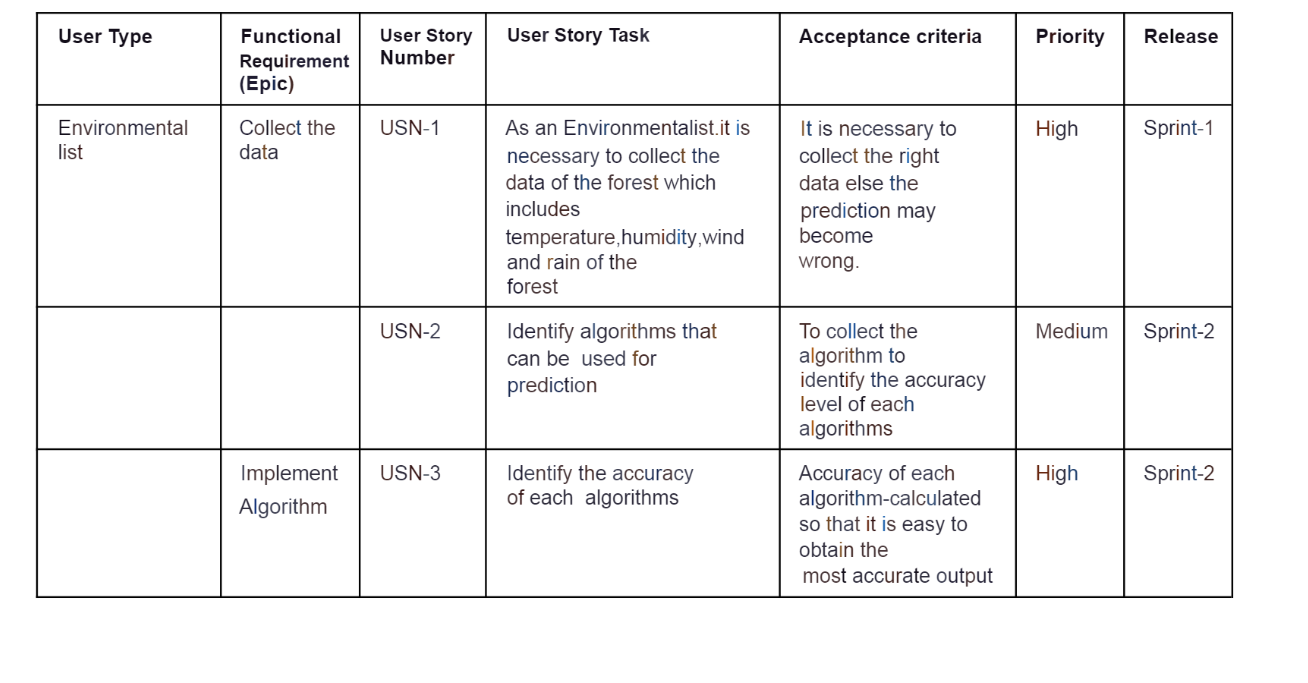
### SOLUTION & TECHNICAL ARCHITECTURE

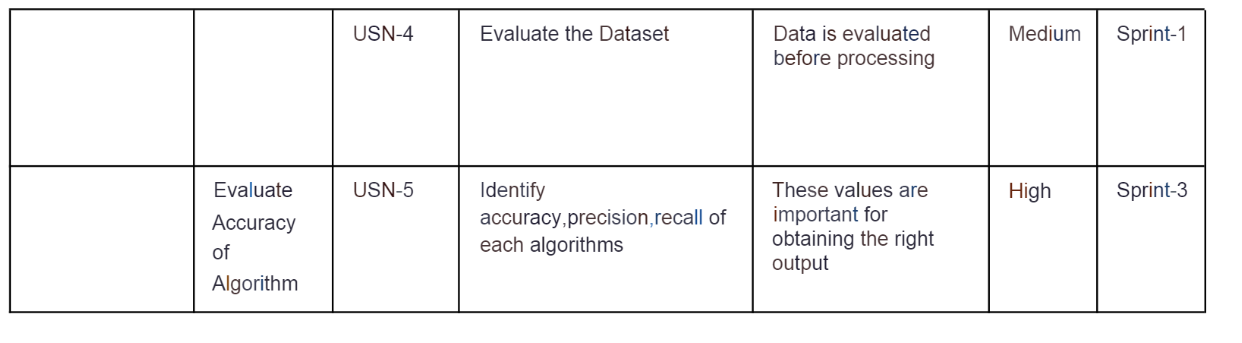






### User Stories:

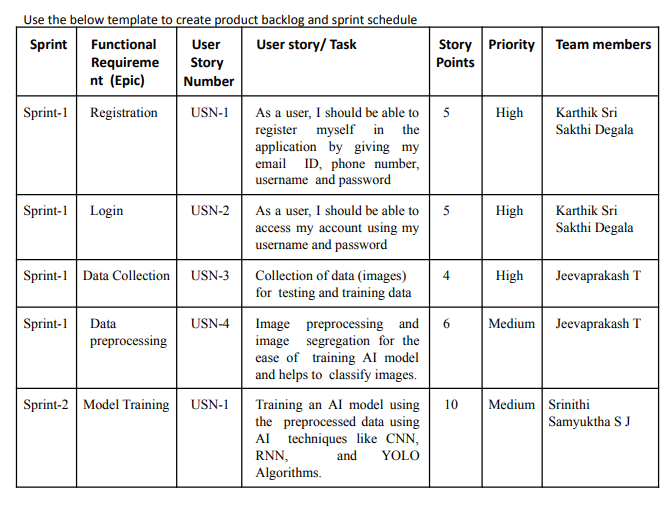
****

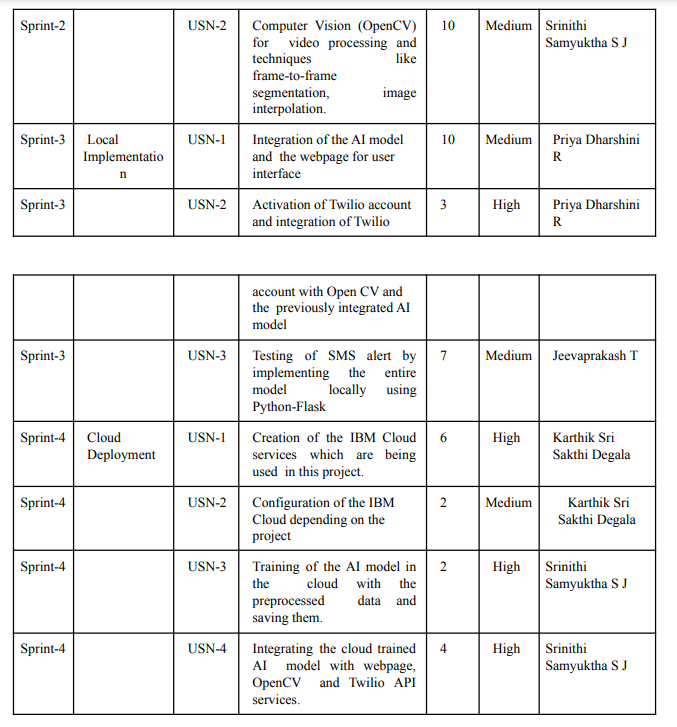
****

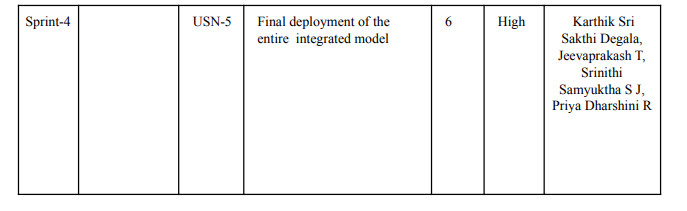
# CHAPTER 6

## PROJECT PLANNING AND SCHEDULING

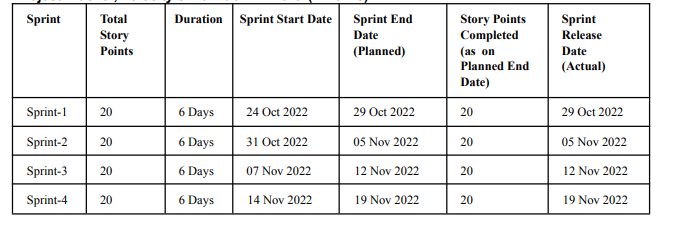
### SPRINT PLANNING AND ESTIMATION

****

****

****

### SPRINT DELIVERY SCHEDULE

****

### Feature 1:

! pip install tensorflow

! pip install opencv-python

# CHAPTER 7

## CODING & SOLUTIONING

! pip install opencv-contrib-python 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 train=ImageDataGenerator(rescale=1./255, shear\_range=0.2,

rotation\_range=180, zoom\_range=0.2, horizontal\_flip=True)

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

train\_dataset = train.flow\_from\_directory("/content/drive/MyDrive/Dataset/train\_set",

target\_size=(128,128), batch\_size = 32, class\_mode = 'binary' )

test\_dataset = test.flow\_from\_directory("/content/drive/MyDrive/Dataset/test\_set",

target\_size=(128,128), batch\_size = 32, class\_mode = 'binary' )

test\_dataset.class\_indices

#to define linear initialisation import sequential from keras.models import Sequential

#to add layer 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()

#add convolutional layer model.add(Convolution2D(32,(3,3),input\_shape=(128,128,3),activation='relu')) #add maxpooling layer

model.add(MaxPooling2D(pool\_size=(2,2))) #add flatten layer

model.add(Flatten()) model.add(Dense(150,activation='relu'))

model.add(Dense(1,activation='sigmoid')) model.compile(loss = 'binary\_crossentropy',

optimizer = "adam", metrics = ["accuracy"])

model.fit\_generator(x\_train,steps\_per\_epoch=14,epochs=5,validation\_data=x\_test,validation\_step s=4)

model.save("/content/drive/MyDrive/archive(1)/forest1.h5") predictions = model.predict(test\_dataset)

predictions = np.round(predictions) predictions

print(len(predictions))

#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

import cv2

#load the saved model

model = load\_model("/content/drive/MyDrive/archive(1)/forest1.h5") def predictImage(filename):

img1 = image.load\_img(filename,target\_size=(128,128)) 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("/content/drive/MyDrive/Dataset/test\_set/with fire/19464620\_401.jpg")

### Feature 2

!pip install tensorflow

!pip install opencv-python

!pip install opencv-contrib-python 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 train=ImageDataGenerator(rescale=1./255,

shear\_range=0.2, rotation\_range=180, zoom\_range=0.2, horizontal\_flip=True)

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

train\_dataset = train.flow\_from\_directory("/content/drive/MyDrive/Dataset/train\_set",

target\_size=(128,128), batch\_size = 32, class\_mode = 'binary' )

test\_dataset = test.flow\_from\_directory("/content/drive/MyDrive/Dataset/test\_set",

target\_size=(128,128), batch\_size = 32, class\_mode = 'binary' )

test\_dataset.class\_indices

#to define linear initialisation import sequential from keras.models import Sequential

#to add layer 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()

#add convolutional layer model.add(Convolution2D(32,(3,3),input\_shape=(128,128,3),activation='relu')) #add maxpooling layer

model.add(MaxPooling2D(pool\_size=(2,2))) #add flatten layer

model.add(Flatten()) model.add(Dense(150,activation='relu'))

model.add(Dense(1,activation='sigmoid')) model.compile(loss = 'binary\_crossentropy',

optimizer = "adam", metrics = ["accuracy"])

model.fit\_generator(x\_train,steps\_per\_epoch=14,epochs=5,validation\_data=x\_test,validation\_s teps=4)

model.save("/content/drive/MyDrive/archive(1)/forest1.h5") predictions = model.predict(test\_dataset)

predictions = np.round(predictions) predictions

print(len(predictions))

#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

import cv2

#load the saved model

model = load\_model("/content/drive/MyDrive/archive(1)/forest1.h5") def predictImage(filename):

img1 = image.load\_img(filename,target\_size=(128,128)) 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("/content/drive/MyDrive/Dataset/test\_set/with fire/19464620\_401.jpg") pip install twilio

pip install playsound #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'/content/drive/MyDrive/archive(1)/forest1.h5') #define video

video = cv2.VideoCapture('/content/Fighting Fire with Fire \_ Explained in 30 Seconds.mp4') #define the features

name = ['forest','with forest'] account\_sid='ACfb4e6d0e7b0d25def63044919f1b96e3' auth\_token='f9ae4fc4a617a527da8672e97eefb2d8' client=Client(account\_sid,auth\_token) message=client.messages \

.create(

body='Forest Fire is detected, stay alert', from\_='+1 302 248 4366',

to='+91 99400 12164'

)

print(message.sid) pip install pygobject def message(val):

if val==1:

from twilio.rest import Client print('Forest fire')

account\_sid='ACfb4e6d0e7b0d25def63044919f1b96e3'

auth\_token='f9ae4fc4a617a527da8672e97eefb2d8' client=Client(account\_sid,auth\_token) message=client.messages \

.create(

body='forest fire is detected, stay alert', #use twilio free number

from\_='+1 302 248 4366',

#to number

to='+91 99400 12164')

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('/content/drive/MyDrive/Dataset/test\_set/with fire/Wild\_fires.jpg',target\_s ize=(128,128))

Y = image.img\_to\_array(img1) x = np.expand\_dims(Y,axis=0) val = model.predict(x) plt.imshow(img1)

plt.show() message(val)

img2 = image.load\_img('/content/drive/MyDrive/Dataset/test\_set/forest/1200px\_Mountainarea.jpg ',target\_size=(128,128))

Y = image.img\_to\_array(img2) x = np.expand\_dims(Y,axis=0) val = model.predict(x) plt.imshow(img2)

plt.show() message(val)

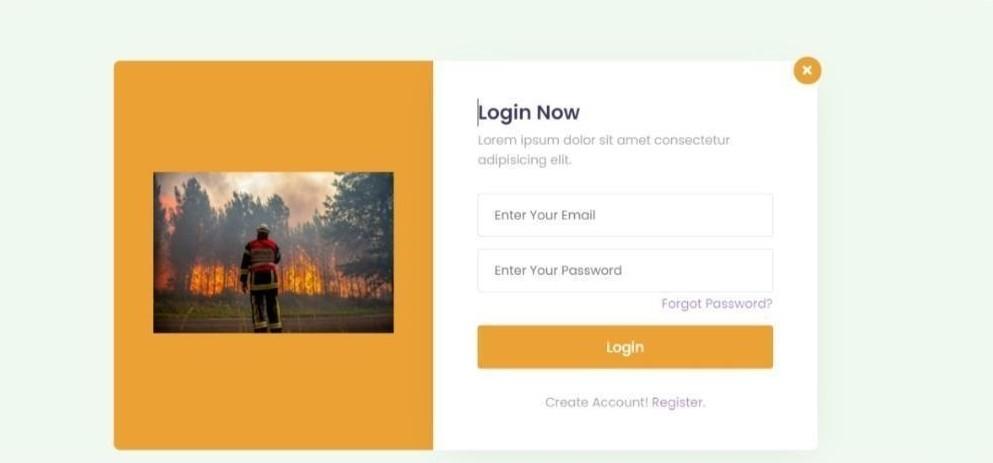
# CHAPTER 8

## TESTING

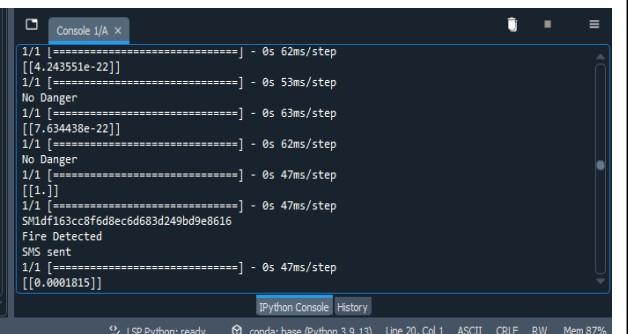
Test cases help guide the tester through a sequence of steps to validate whether

a software application is free of bugs, and working as required by the end-user. Learning how to write test cases for software requires basic writing skills, an attention to detail, and a good understanding of the application under test (AUT).

### TEST CASES



### User Acceptance Testing:



# CHAPTER 9

## RESULTS

### Performance Metrics:



# CHAPTER 10

## ADVANTAGES & DISADVANTAGES

### ADVANTAGES

* + - Avoid Smoke Inhalation. The most important reason is perhaps the only one you really need.
    - Early Detection. The earlier a fire is detected, the faster it will be that firefighters will respond.
    - Insurance Discounts.
    - 24/7 Monitoring.
    - Easy & Affordable

### DISADVANTAGES

* + - The system is essentially useless if the batteries aren't charged, since it won't work properly.
    - There is a bit of a burden to business owners to always remember to keep the batteries fresh so the system operates properly when you need it most.

**CHAPTER 11**

## CONCLUSION

This project will help in early detection of forest ﬁre and the prevention. It also involves the risk factor of analyzing the drone images of affected areas using machine learning algorithm which overcomes the existing project. This system detects the ﬁre conditions in a short time before any ﬁre accidents spreads over the forest area. The scope of using video frames in the detection of ﬁre using machine learning is challenging as well as innovative. If this system with less error rate can be implemented at a large scale like in big factories, houses, forests, it is possible to prevent damage and loss due to random ﬁre accidents by making use of the Surveillance System.

# CHAPTER 12

## FUTURE SCOPE

Future Scope In future, we are planning to install smart water tank system in dense forest where reachability of resources and ﬁreﬁghters is diﬃcult. In addition to that we will be updating the system with more features and reliability. We will also include a high pitch sound system that will keep away the animals from the site of ﬁre.The proposed system can be developed to more advanced system by integrating wireless sensors with CCTV for added protection and precision. The algorithm shows great promise in adapting to various environment.

## APPENDIX

### Source Code:

#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'forest1.h5') #define video

video = cv2.VideoCapture(0) #define the features

name = ['forest','with forest']

account\_sid = 'AC557b4c7a685d072baa73125f61031af3' auth\_token = 'a59cd5e5fdfddcc9ab008273557f8f78' client = Client(account\_sid, auth\_token)

message = client.messages \

.create(

body='Forest fire is detected , stay alert', from\_='+14247991869', to='+918940722793'

)

print(message.sid) #import opencv library import cv2

22

#import numpy

import numpy as np

#import images and load\_model function from keras from keras\_preprocessing import image

from keras.models import load\_model #import client from twilio API

from twilio.rest import Client #import playsound package

from playsound import playsound #load the saved model

model = load\_model(r'forest1.h5') video = cv2.VideoCapture(0) name = ['forest','with fire'] while(1):

success, frame=video.read() cv2.imwrite("image.jpg",frame) img=image.load\_img("image.jpg",target\_size=(128,128,3)) x=image.img\_to\_array(img)

x=np.expand\_dims(x,axis=0) pred=model.predict(x) p=pred[0]

print(pred)

##cv2.putText(frame,"predicted class= "+str(name[p]), (100,100), ## cv2.FONT\_HERSHEY\_SIMPLEX, 1, (0,0,0), 1)

pred=model.predict(x) if pred[0]==1:

account\_sid = 'AC557b4c7a685d072baa73125f61031af3' auth\_token = 'a59cd5e5fdfddcc9ab008273557f8f78' client = Client(account\_sid, auth\_token) message=client.messages\

.create( 23

body='Forest Fire is Detected, stay alert', from\_='+14247991869',to='+918940722793')

print(message.sid) print('Fire Detected') print('SMS sent')

playsound(r'C:\Users\My\Downloads\buzzer.mp3') else:

print("No Danger") cv2.imshow("image",frame)

if cv2.waitKey(1) & 0xFF ==ord('a'): break

video.release() cv2.destroyAllWindows()

### GITHUB

### **https://github.com/IBM-EPBL/IBM-Project-14233-1659546538**

**PROJECT DEMO**

<https://drive.google.com/drive/folders/12eRdGd7rbR8L_V3kRFJ1wIx1QLsfmzd6>