

# **PROJECT DEVELOPMENT PHASE**

## **SPRINT-III**

### **MODEL BUILDING**

[Click Here to view the Project\(Hyperlink\)](#)

<b>Date</b>	04 November 2022
<b>Team ID</b>	PNT2022TMID49492
<b>Project Name</b>	Emerging Methods for Early Detection of Forest Fires
<b>MaximumMarks</b>	4 Marks

*Import model building libraries* #import keras  
libraries import numpy as np import tensorflow  
from tensorflow.keras.models import Sequential  
from tensorflow.keras import layers from  
keras.layers import Dense from keras.layers  
import Conv2D from keras.layers import  
MaxPooling2D,Dropout from keras.layers import  
Flatten

### ***Initializing the model***

```
model=Sequential() Add
```

### ***CNN 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())
```

### ***Add Hidden Layer***

```
#add hidden layer
```

```
model.add(Dense(units=128,activation='relu'))
```

```
#add output layer
```

```
model.add(Dense(units=46,activation='softmax'))
```

### ***Configure the learning process***

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

### ***Train the model***

```
model.fit(x_train,epochs=10,steps_per_epochs=len((x_train)
```

```
from google.colab import drive drive.mount('/content/drive')
```

### ***Save The Model***

```
model.save("forestwithfire.h5")
```

### ***Predictions***

```
# 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
```

```
#load the saved model model =  
load_model("forestwithfire.h5")
```

```
#give any random image path
```

```
img=image.load_img(r'/content/drive/MyDrive/DataCollection/training/F  
orest with fire/with fire (10).jpg') x = image.img_to_array(img)  
res = cv2.resize(x,dsize=(128,128),interpolation=cv2.INTER_CUBIC)
```

```
#expand the image shape
```

```
x=np.expand_dims(res,axis=0)  
pred= model.predict(x_train)  
pred
```