PROJECT DEVELOPMENT PHASE

SPRINT-3

Team ID	PNT2022TMID21552
Project Name	EMERGING METHODS FOR EARLY DETECTION OF FOREST FIRES

In this sprint, we have trained the model on IBM Watson and used it locally for predictions. Two simple html pages has also been developed. One for home and one for uploading image

Website code:

Home.html

```
<!DOCTYPE html>
<html lang="en">
<head>
 <meta charset="UTF-8">
 <meta http-equiv="X-UA-Compatible" content="IE=edge">
 <meta name="viewport" content="width=device-width, initial-scale=1.0">
 <title>Simple HTML HomePage</title>
 <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">
 <style>
  @import url('https://fonts.googleapis.com/css2?family=Sriracha&display=swap');

  body {
   margin: 0;
   box-sizing: border-box;
  }

  /* CSS for header */
  .header {
   display: flex;
   justify-content: space-between;
```

```
  align-items: center;
  background-color: #f5f5f5;
}

.header .logo {
  font-size: 25px;
  font-family: 'Sriracha', cursive;
  color: #000;
  text-decoration: none;
  margin-left: 30px;
}

.nav-items {
  display: flex;
  justify-content: space-around;
  align-items: center;
  background-color: #f5f5f5;
  margin-right: 20px;
}

.nav-items a {
  text-decoration: none;
  color: #000;
  padding: 35px 20px;
}
/* CSS for main element */
.intro {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  width: 100%;
```

```
    height: 520px;
    background: linear-gradient(to bottom, rgba(0, 0, 0, 0.5) 0%, rgba(0, 0, 0, 0.5) 100%),
url("https://magazine.columbia.edu/sites/default/files/styles/full_width_card/public/2018-
09/Wild-fires.jpg?itok=8pfGkL7z");
    background-size: cover;
    background-position: center;
    background-repeat: no-repeat;
  }

  .intro h1 {
    font-family: sans-serif;
    font-size: 60px;
    color: #fff;
    font-weight: bold;
    text-transform: uppercase;
    margin: 0;
  }

  .intro p {
    font-size: 20px;
    color: #d1d1d1;
    text-transform: uppercase;
    margin: 20px 0;
  }

  .intro button {
    background-color: #5edaf0;
    color: #000;
    padding: 10px 25px;
    border: none;
    border-radius: 5px;
    font-size: 20px;
```

```
  font-weight: bold;
  cursor: pointer;
  box-shadow: 0px 0px 20px rgba(255, 255, 255, 0.4)
}

.achievements {
  display: flex;
  justify-content: space-around;
  align-items: center;
  padding: 40px 80px;
}

.achievements .work {
  display: flex;
  flex-direction: column;
  justify-content: center;
  align-items: center;
  padding: 0 40px;
}

.achievements .work i {
  width: fit-content;
  font-size: 50px;
  color: #333333;
  border-radius: 50%;
  border: 2px solid #333333;
  padding: 12px;
}

.achievements .work .work-heading {
  font-size: 20px;
  color: #333333;
```

```
  text-transform: uppercase;
  margin: 10px 0;
}

.achievements .work .work-text {
  font-size: 15px;
  color: #585858;
  margin: 10px 0;
}

/* CSS for footer */
.footer {
  display: flex;
  justify-content: space-between;
  align-items: center;
  background-color: #302f49;
  padding: 40px 80px;
}

.footer .copy {
  color: #fff;
}

.bottom-links {
  display: flex;
  justify-content: space-around;
  align-items: center;
  padding: 10px 0;
}

.bottom-links .links {
  display: flex;
```

```
    flex-direction: column;
    justify-content: center;
    align-items: center;
    padding: 0 120px;
   }

   .bottom-links .links span {
    font-size: 20px;
    color: #fff;
    text-transform: uppercase;
    margin: 20px 0;
   }

   .bottom-links .links a {
    text-decoration: none;
    color: #a1a1a1;
    padding: 10px 120px;
   }
  </style>
</head>

<body>
  <header class="header">
   <a href="#" class="logo">Forest Fire Detection</a>
   <nav class="nav-items">
   <a class="active" href="home.html">Home</a>
   <a   href="detect.html">Detect</a>
   </nav>
  </header>
  <main>
   <div class="intro">
    <h1>Emerging Methods For Early Detection Of Forest Fire</h1>
```

```
    <p>This is a website that helps in early detection of fires inorder to help preserve wildlife.</p>


   </div>

<h1><center>DID YOU KNOW ? <center></h1>

   <div class="achievements">


    <div class="work">


     <p class="work-text">Since 1983, the National Interagency Fire Center has documented an average of approximately 70,000 wildfires per year.</p>

    </div>

    <div class="work">


     <p class="work-text">The largest forest fire was the Miramichi Fire in New Brunswick, Canada in 1825 which destroyed about 3 million acres (12150 km$^2$). this fire killed 160 people.</p>

    </div>

    <div class="work">


     <p class="work-text"> At least 60 per cent of districts in India are affected by forest fires each year</p>

    </div>

   </div>


 <footer class="footer">

   <div class="bottom-links">

    <div class="links">

     <span>DEVELOPED BY</span>

     <a href="#">Monisha M</a>

     <a href="#">Nishanthini V</a>

     <a href="#">Reenasree S</a>

         <a href="#">Shruti S</a>

         <a href="#">Sreeram B</a>
```

```
    </div>
    <div class="links">
      <span>Social Links</span>
      <a href="#"><i class="fab fa-facebook"></i></a>
      <a href="#"><i class="fab fa-twitter"></i></a>
      <a href="#"><i class="fab fa-instagram"></i></a>
    </div>
  </div>
 </footer>
</body>
</html>
```

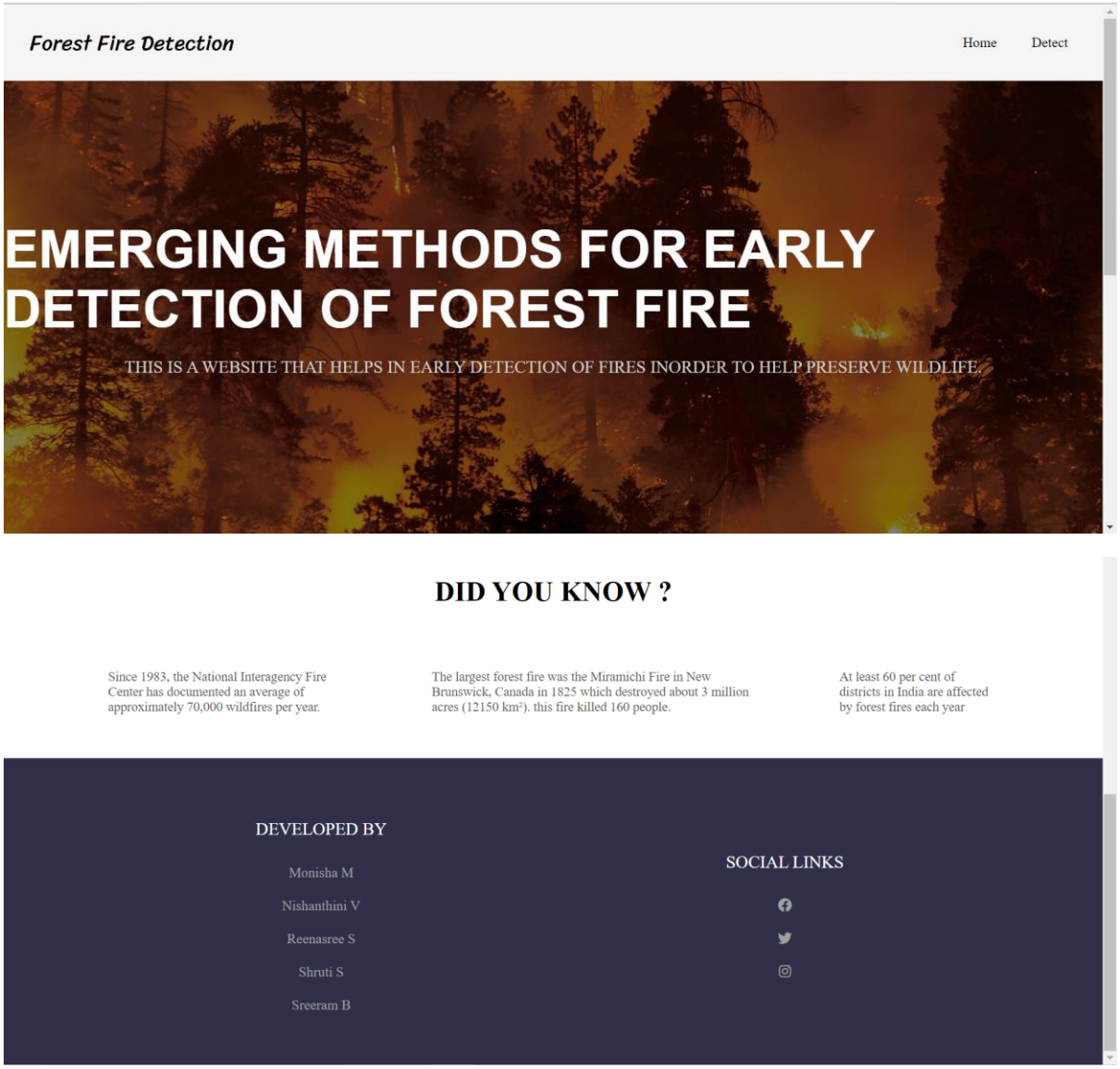
Index.html

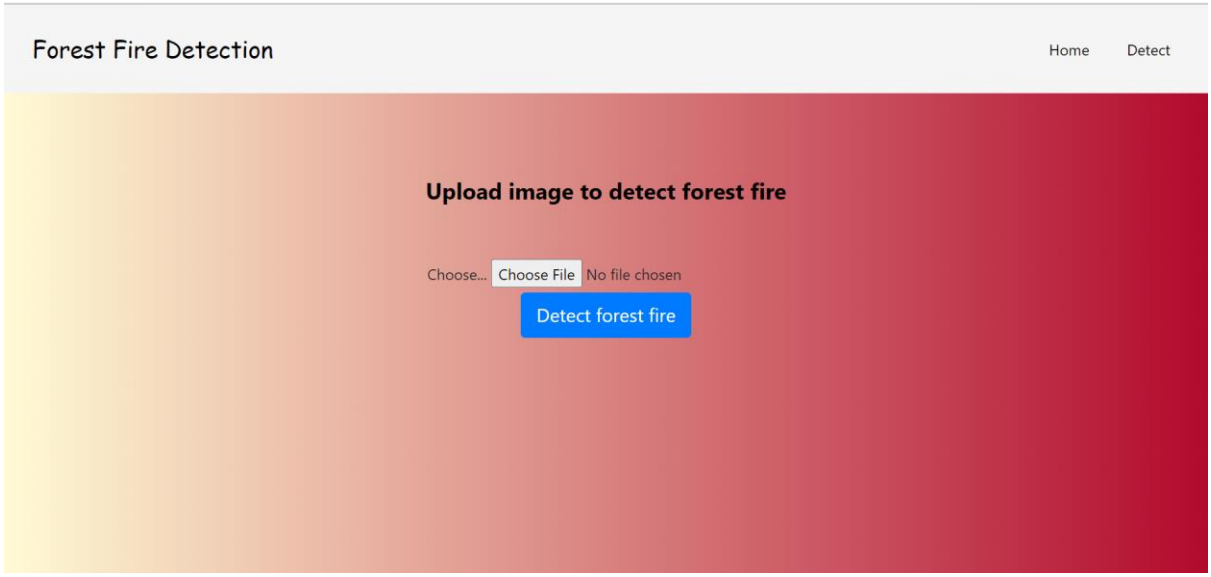
```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <meta http-equiv="X-UA-Compatible" content="ie=edge">
   <title>Predict</title>
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">
   <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
   <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js"></script>
   <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js"></script>
   <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js"></script>
   <link href="{{ url_for('static', filename='css/main.css') }}" rel="stylesheet">
<style>
body
{
   background-image: linear-gradient(to right, #fffbd5, #b20a2c);
   background-size: cover;
```

```
}
.header {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #f5f5f5;
  }

  .header .logo {
    font-size: 25px;
    font-family: 'Sriracha', cursive;
    color: #000;
    text-decoration: none;
    margin-left: 30px;
  }

  .nav-items {
    display: flex;
    justify-content: space-around;
    align-items: center;
    background-color: #f5f5f5;
    margin-right: 20px;
  }

  .nav-items a {
    text-decoration: none;
    color: #000;
    padding: 35px 20px;
  }
.bar
{
margin: 0px;
```

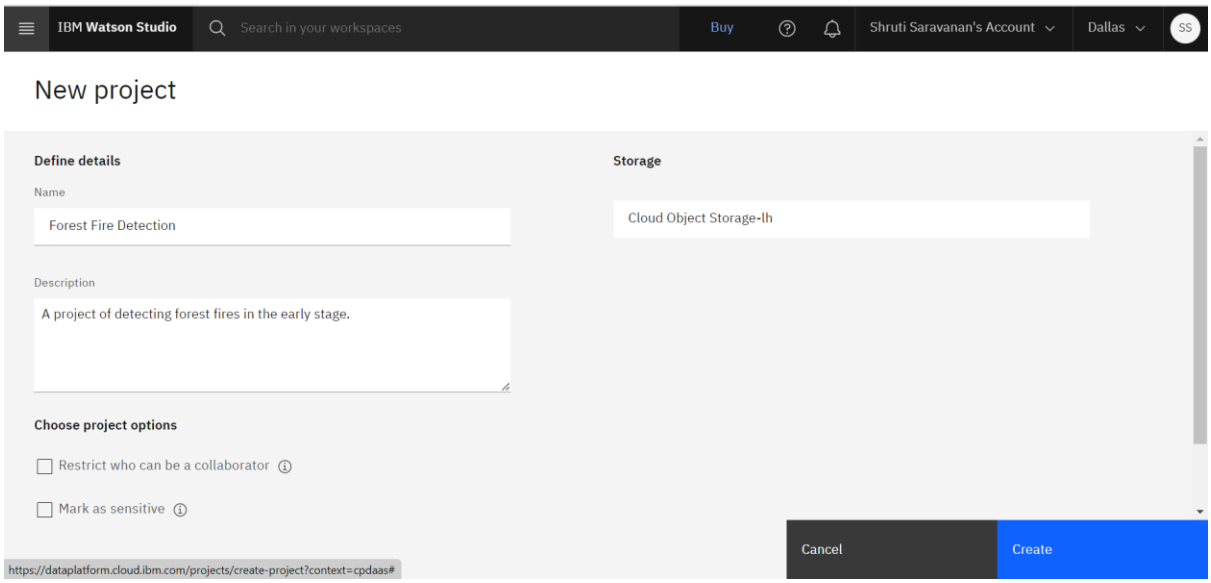
```
padding:20px;
background-color:white;
opacity:0.6;
color:black;
font-family:'Roboto',sans-serif;
font-style: italic;
border-radius:20px;
font-size:25px;
}
a
{
color:grey;
float:right;
text-decoration:none;
font-style:normal;
padding-right:20px;
}
.div{
  background-color: lightgrey;
  width: 500px;
  border: 10px solid peach;
  padding: 20px;
  margin: 20px;
  height: 500px;
}
</style>
</head>
<body>
  <header class="header">
    <a href="#" class="logo">Forest Fire Detection</a>
    <nav class="nav-items">
    <a class="active" href="home.html">Home</a>
```

```
   <a   href="detect.html">Detect</a>
   </nav>
 </header>
<br>
<div>
<br>
<br>
<h1><center><font color="black" size="5" ><b>Upload image to detect forest fire
</center></b></font></h1><br><br>
<div style="margin-left: 450px;">
   <form id="upload-file" method="post" enctype="multipart/form-data">
     <label for="imageUpload" class="upload-label">
        Choose...
      </label>
      <input type="file" name="file" id="imageUpload" accept=".png, .jpg, .jpeg">
   </form>
  <center> <div class="image-section" style="display:none;">
      <div class="img-preview">
        <div id="imagePreview">
        </div></center>
      </div>
      <center><div>
        <button type="button" class="btn btn-primary btn-lg " id="btn-predict">Detect forest
fire</button>
     </center></div>
   </div>
</div>
</html>
```

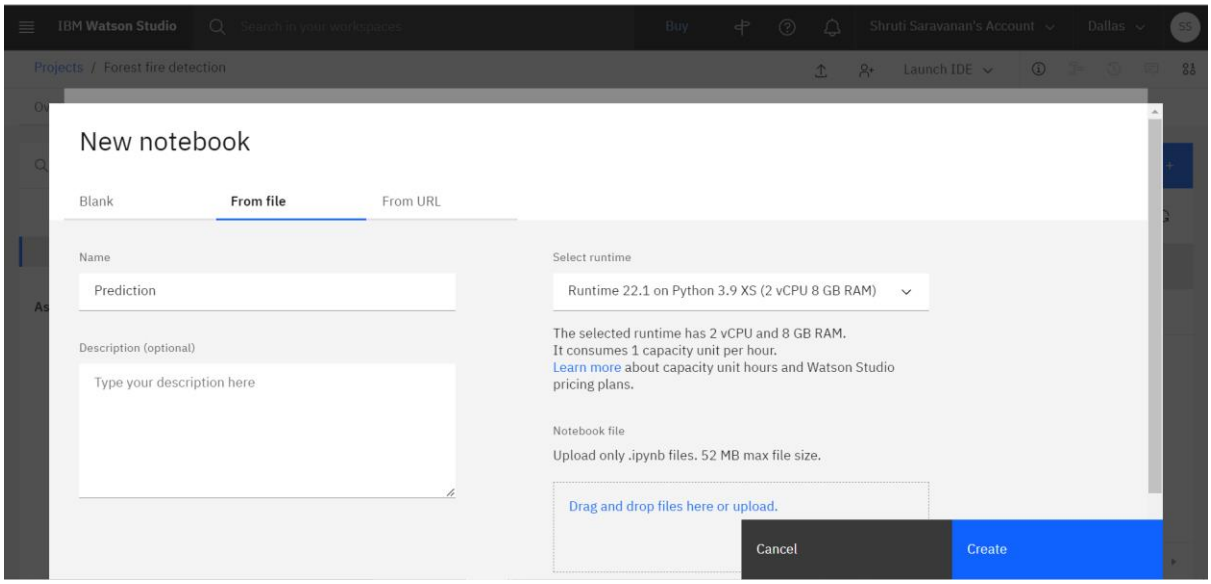




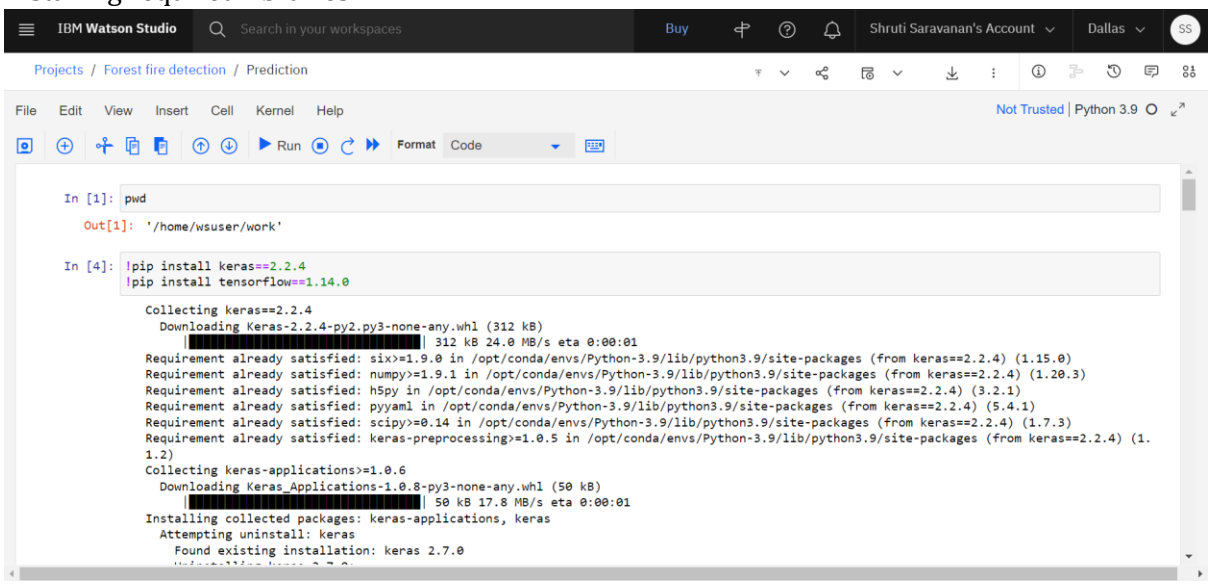
Creating a new project in IBM Watson:



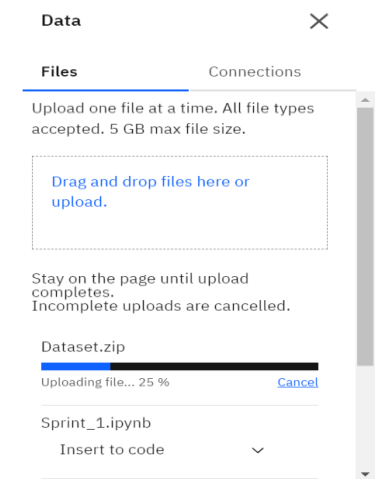
Starting a Jupyter notebook:

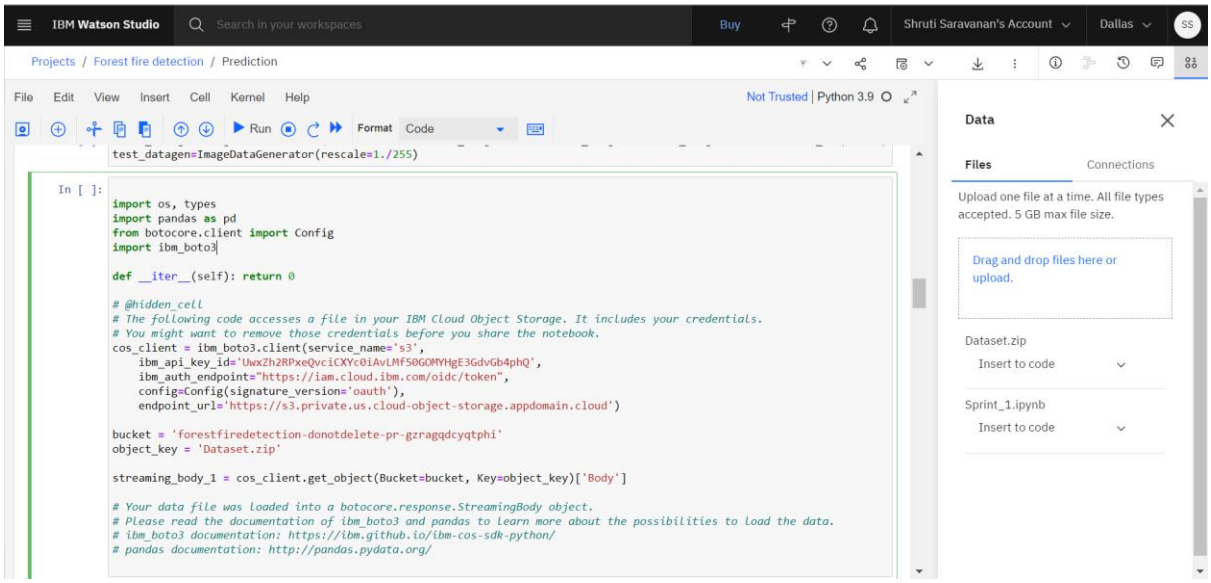


Installing required libraries



Uploading dataset and integrating to code:





```
In [*]: from io import BytesIO
        import zipfile
        unzip=zipfile.ZipFile(BytesIO(streaming_body_1.read()),'r')
        file_paths = unzip.namelist()
        for path in file_paths:
            unzip.extract(path)
```

```
In [17]: import os
         file= os.listdir('/home/wsuser/work/Dataset/train_set')

In [19]: x_train=train_datagen.flow_from_directory('/home/wsuser/work/Dataset/train_set',target_size=(128,128),batch_size=32,clas

         Found 436 images belonging to 2 classes.

In [20]: x_test=test_datagen.flow_from_directory('/home/wsuser/work/Dataset/train_set',target_size=(128,128),batch_size=32,class_

         Found 436 images belonging to 2 classes.

In [21]: x_train.class_indices
Out[21]: {'forest': 0, 'with fire': 1}
```

Building CNN model:

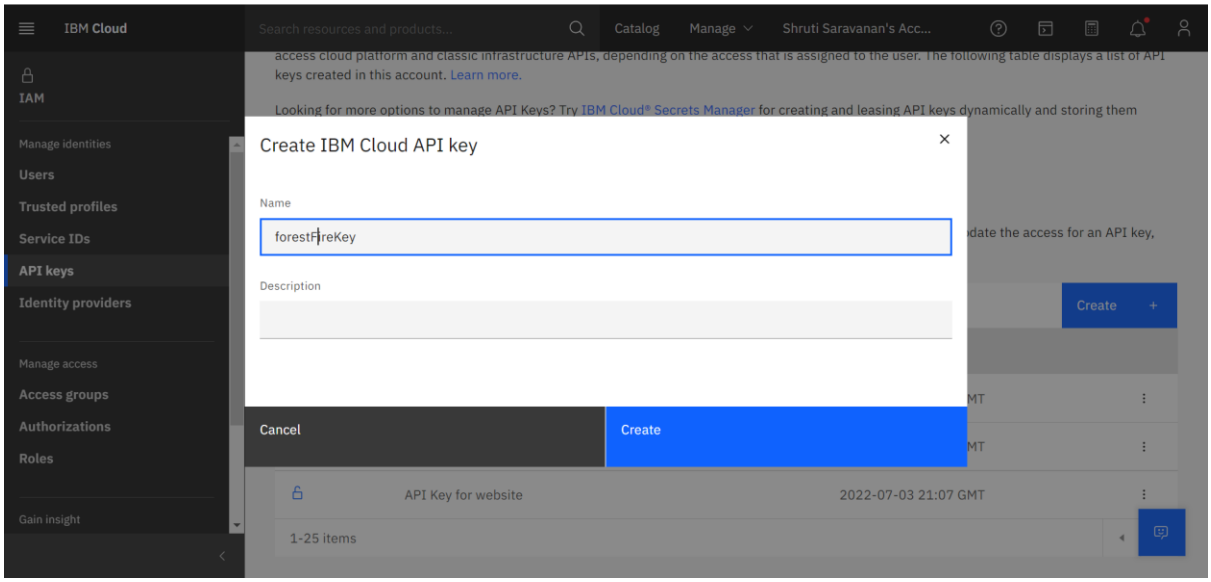
```
In [22]: model=Sequential()

In [23]: model.add(Convolution2D(32,(3,3),input_shape=(128,128,3),activation='relu'))

In [24]: model.add(MaxPooling2D(pool_size=(2,2)))

In [25]: model.add(Flatten())

In [26]: model.add(Dense(150,activation='relu'))

In [27]: model.add(Dense(1,activation='sigmoid'))

In [28]: model.compile(loss='binary_crossentropy',optimizer="adam",metrics=["accuracy"])
```

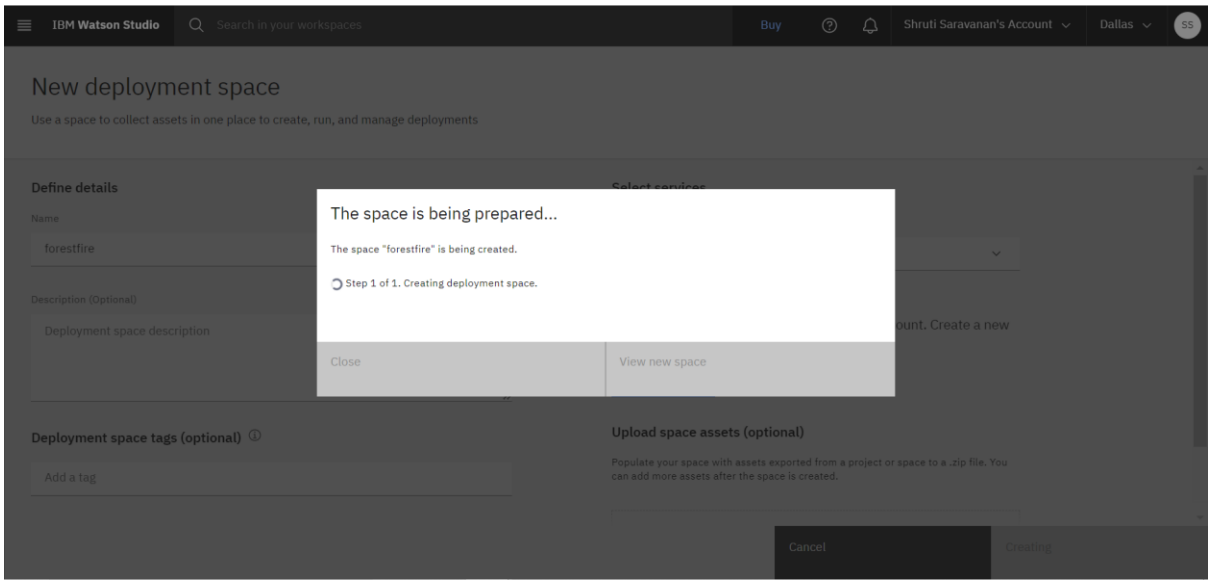
```
In [29]: y=model.fit_generator(x_train,steps_per_epoch=14,epochs=10,validation_data=x_test,validation_steps=4)

         Epoch 1/10
         14/14 [==============================] - 20s 1s/step - loss: 2.0573 - accuracy: 0.7156 - val_loss: 0.9358 - val_accu
         racy: 0.8281
         Epoch 2/10
         14/14 [==============================] - 19s 1s/step - loss: 0.4683 - accuracy: 0.8601 - val_loss: 1.9996 - val_accu
         racy: 0.7656
         Epoch 3/10
         14/14 [==============================] - 19s 1s/step - loss: 0.3691 - accuracy: 0.8807 - val_loss: 0.5620 - val_accu
         racy: 0.8359
         Epoch 4/10
         14/14 [==============================] - 20s 1s/step - loss: 0.4316 - accuracy: 0.8647 - val_loss: 0.8414 - val_accu
         racy: 0.7812
         Epoch 5/10
         14/14 [==============================] - 19s 1s/step - loss: 0.2996 - accuracy: 0.9014 - val_loss: 0.1518 - val_accu
         racy: 0.9531
         Epoch 6/10
         14/14 [==============================] - 19s 1s/step - loss: 0.2202 - accuracy: 0.9083 - val_loss: 0.3765 - val_accu
         racy: 0.8984
         Epoch 7/10
         14/14 [==============================] - 18s 1s/step - loss: 0.1914 - accuracy: 0.9335 - val_loss: 0.5691 - val_accu
         racy: 0.8359
         Epoch 8/10
         14/14 [==============================] - 18s 1s/step - loss: 0.3367 - accuracy: 0.8693 - val_loss: 0.3724 - val_accu
         racy: 0.8672
         Epoch 9/10
```

```
In [30]: model.save("forests.h5")

In [32]: !tar -zcvf fire-detection_model.tgz forests.h5

         forests.h5

In [35]: !zip forest-fire-detection_model.zip forests.h5

           adding: forests.h5 (deflated 10%)

In [36]: ls -1

         Dataset/
         fire-detection_model.tgz
         forest-fire-detection_model.zip
         forests.h5
         zcvf.zip
```

In [39]:
```
!pip install watson-machine-learning-client --upgrade
```

```
Collecting watson-machine-learning-client
  Downloading watson_machine_learning_client-1.0.391-py3-none-any.whl (538 kB)
     |████████████████████████████████| 538 kB 20.7 MB/s eta 0:00:01
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine
-learning-client) (1.3.4)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machin
e-learning-client) (2022.9.24)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine
-learning-client) (0.3.3)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machi
ne-learning-client) (2.26.0)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machin
e-learning-client) (1.26.7)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machi
ne-learning-client) (0.8.9)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-ma
chine-learning-client) (2.11.0)
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-
learning-client) (1.18.21)
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-l
earning-client) (4.62.3)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (fro
m boto3->watson-machine-learning-client) (0.10.0)
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (f
rom boto3->watson-machine-learning-client) (0.5.0)
```





In [77]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials={
    "url":"https://us-south.ml.cloud.ibm.com",
    "apikey":"eTqOW491DNxgOkkbANwAhjl5OlNp0HQs1oSSAXDFD4qJ"
}
client=APIClient(wml_credentials)
```

In [78]:
```
client=APIClient(wml_credentials)
```

In [79]:
```
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'])
```

In [80]:
```
space_uid=guid_from_space_name(client,'forestfire')
print("Space UID ="+ space_uid)
```

```
Space UID =b9194455-9cab-4668-9eb4-e15264d2e823
```

In [81]:
```
client.set.default_space(space_uid)
```

Out[81]: 'SUCCESS'

:
```
client.software_specifications.list(100)
```

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
pytorch_1.1-py3.6	10ac12d6-6b30-4ccd-8392-3e922c096a92	base
tensorflow_1.15-py3.6-ddl	111e41b3-de2d-5422-a4d6-bf776828c4b7	base

In [92]:
```
software_spec_uid=client.software_specifications.get_uid_by_name("tensorflow_2.1-py3.6")
software_spec_uid
```

Out[92]: 'Not Found'

In [ ]:
```
model_details = client.repository.store_model(model='model_fire_detection.tgz'
                                              ,meta_props={
    client.repository.ModelMetaNames.NAME:"FireCNN",
    client.repository.ModelMetaNames.TYPE:"keras_2.7.0",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid
})
model_id=client.repository.get_model_uid(model_details)
```

In [ ]:
```
model_id
```

In [94]:
```
from keras.models import load_model
from keras.preprocessing import image
```

In [95]:
```
model = load_model('forests.h5')
```

In [106]:
```
from keras.models import load_model
import cv2
import numpy as np
from PIL import Image
from tensorflow.keras.utils import img_to_array
import matplotlib.pyplot as plt
from twilio.rest import Client
from playsound import playsound
model = load_model('forests.h5')
account_sid='AC33e4f23328753859047817ac8815083b'
auth_token ='ec85f2a8b7e067400404fd9c0c565797'
client=Client(account_sid,auth_token)
def prediction(img_path):
    i = cv2.imread(img_path)
    i = cv2.cvtColor(i, cv2.COLOR_BGR2RGB)
    img = Image.open(img_path)
    img = img.resize((128,128))
    x = img_to_array(img)
    x = np.expand_dims(x,axis=0)
    pred = model.predict(x)
    plt.imshow(i)
    if(pred==[[1.]]) :
      message=client.messages \
      .create(
          body='FOREST FIRE IS DECTECTED IN AREA,stay alert',
          #use twilio free number
          from_='+12535288281',
          #to number
           from_='+12535288281',
           #to number
           to='+918610505460')
      print('Fire Detected')
      print('SMS sent!')
    else:
      print("NO FOREST FIRE DETECTED")
      print("no message sent")
```

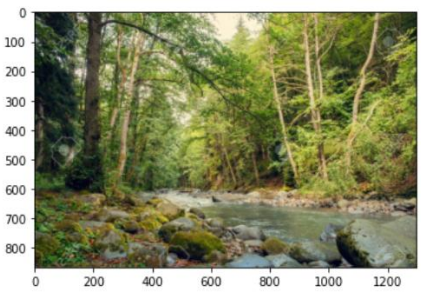
In [37]:
```
prediction('/home/wsuser/work/test_set/forest/1170x500_Ireland_web.jpg')
```

```
1/1 [==============================] - 0s 31ms/step
NO FOREST FIRE DETECTED
no message sent
```

Using model locally:

In [35]:
```
prediction(r'/content/gdrive/My Drive/Dataset/test_set/forest/111188170_river_in_the_mountain_forest.jpg')
```

```
1/1 [==============================] - 0s 32ms/step
NO FOREST FIRE DETECTED
no message sent
```



In [36]:
```
prediction(r'/content/gdrive/My Drive/Dataset/test_set/with fire/RED7_May29_1986.jpg')
```

```
1/1 [==============================] - 0s 30ms/step
NO FOREST FIRE DETECTED
no message sent
```