

SPRINT – 3

APPLICATION BUILDING

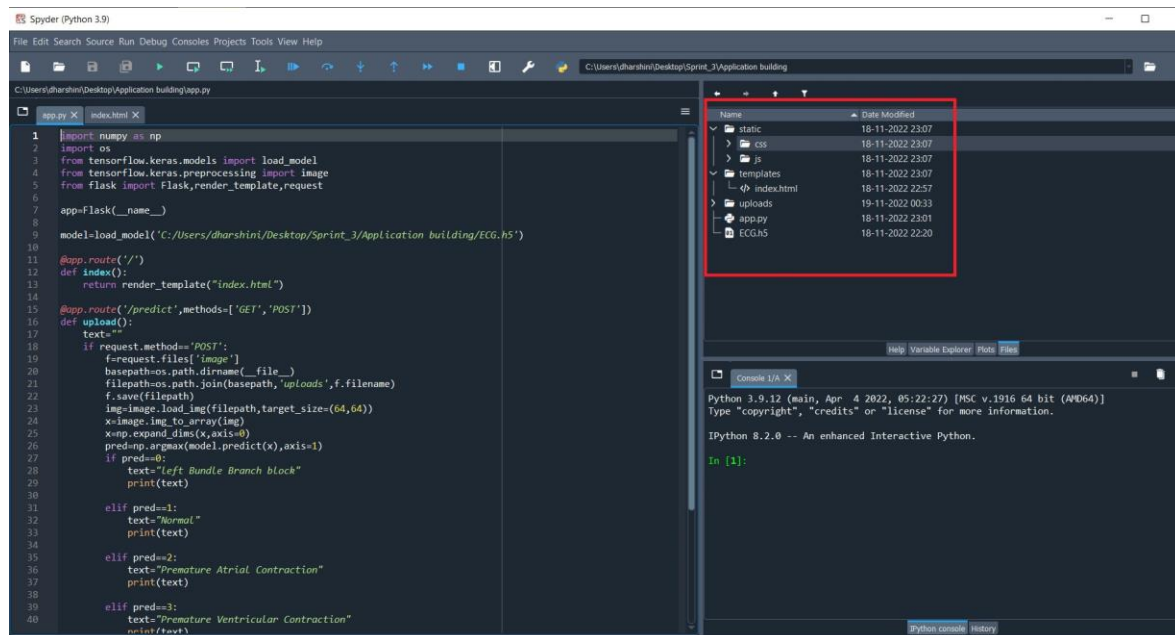
BUILD THE PYTHON CODE

Date	15 November 2022
Team ID	PNT2022TMID03593
Project Name	Project - Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral ImageRepresentation
Sprint	3

TASK:

Build the Python code.

PROJECT STRUCTURE:



App.py:

```
import numpy as np
import os

from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
from flask import Flask,render_template,request

app=Flask(__name__)

model=load_model('C:/Users/dharshini/Desktop/Sprint_3/Application building/ECG.h5')

@app.route('/')
def index():
    return render_template("index.html")

@app.route('/predict',methods=['GET','POST'])
def upload():
    text=""
    if request.method=='POST':
        f=request.files['image']
        basepath=os.path.dirname(_file_)
        filepath=os.path.join(basepath,'uploads',f.filename)
        f.save(filepath)
        img=image.load_img(filepath,target_size=(64,64))
        x=image.img_to_array(img)
        x=np.expand_dims(x,axis=0)
        pred=np.argmax(model.predict(x),axis=1)
        if pred==0:
```

```
        text="left Bundle Branch
        block"print(text)

elif pred==1:
    text="Normal"
    print(text)

elif pred==2:
    text="Premature Atrial Contraction"
    print(text)

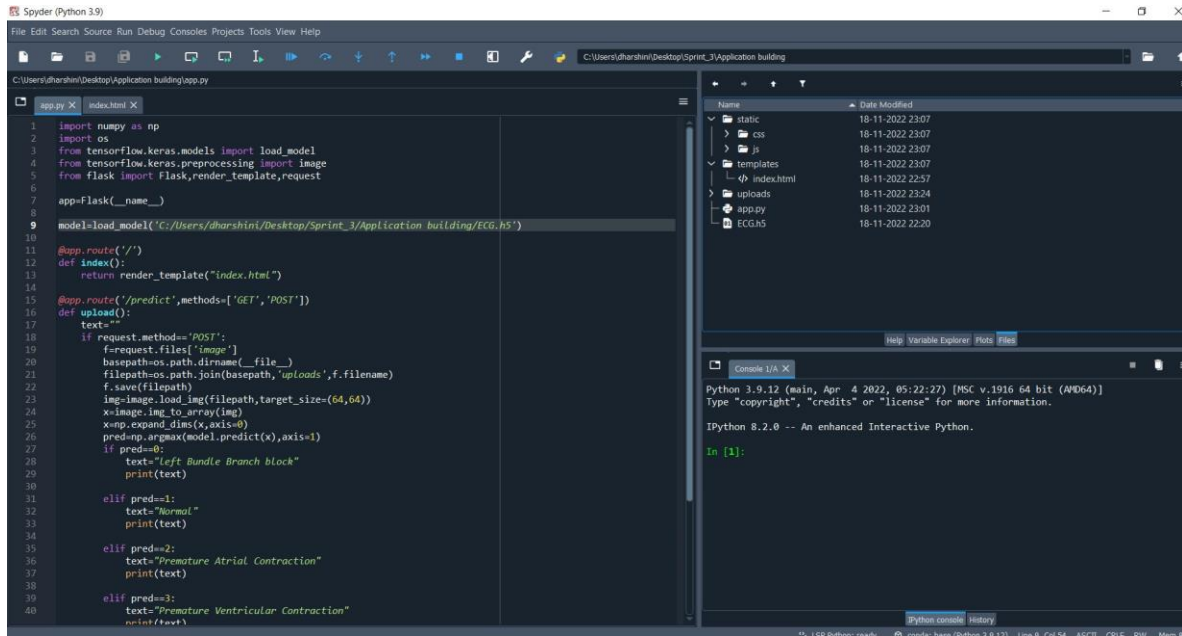
elif pred==3:
    text="Premature Ventricular Contraction"
    print(text)

elif pred==4:
    text="Right Bundle Branch
    Block"print(text)
else:
    text="Ventricular
    Fibrillation"print(text)

return text

if __name__=='_main_':
    app.run(debug=False)
```

APP.PY(SCREEN SHOT):



The screenshot displays the Spyder Python IDE interface. The main editor window shows the file `app.py` with the following Python code:

```
1 import numpy as np
2 import os
3 from tensorflow.keras.models import load_model
4 from tensorflow.keras.preprocessing import image
5 from flask import Flask, render_template, request
6
7 app=Flask(__name__)
8
9 model=load_model('C:/Users/dharshini/Desktop/Sprint_3/Application building/ECG.h5')
10
11 @app.route('/')
12 def index():
13     return render_template("index.html")
14
15 @app.route('/predict', methods=['GET', 'POST'])
16 def upload():
17     text=""
18     if request.method=="POST":
19         f=request.files['image']
20         filepath=os.path.dirname(__file__)
21         filepath=os.path.join(filepath, 'uploads', f.filename)
22         f.save(filepath)
23         img=image.load_img(filepath, target_size=(64,64))
24         x=image.img_to_array(img)
25         x=np.expand_dims(x,axis=0)
26         pred=np.argmax(model.predict(x),axis=1)
27         if pred==0:
28             text="left Bundle Branch block"
29             print(text)
30         elif pred==1:
31             text="Normal"
32             print(text)
33         elif pred==2:
34             text="Premature Atrial Contraction"
35             print(text)
36         elif pred==3:
37             text="Premature Ventricular Contraction"
38             print(text)
39
40
```

The right sidebar shows the file explorer with the following structure:

- static (18-11-2022 23:07)
- css (18-11-2022 23:07)
- js (18-11-2022 23:07)
- templates (18-11-2022 23:07)
 - index.html (18-11-2022 22:57)
- uploads (18-11-2022 23:24)
- app.py (18-11-2022 23:01)
- ECG.h5 (18-11-2022 22:20)

The bottom console window shows the Python 3.9.12 environment and the IPython 8.2.0 prompt, indicating the application is running.