

PROJECT DEVELOPMENT PHASE

SPRINT 1

Team ID	PNT2022TMID38970
Project Name	Project – Signs with Smart Connectivity for Better Road Safety

PROGRAM CODE :

1. Weather.py

This file contains a utility function that uses the OpenWeather API to retrieve the weather. Only a few of the necessary API response parameters are returned.

Python code

```
import requests as reqs
```

```
def get(myLocation,APIKEY):
```

```
    apiURL =
```

```
f"https://api.openweathermap.org/data/2.5/weather?q={myLocation}&appid={API  
KEY}"
```

```
    responseJSON = (reqs.get(apiURL)).json()
```

```
    returnObject = {
```

```
        "temperature" : responseJSON['main']['temp'] - 273.15,
```

```
        "weather" : [responseJSON['weather'][_]['main'].lower() for _ in  
range(len(responseJSON['weather']))],
```

```
        "visibility" : responseJSON['visibility']/100, # visibility in percentage where  
10km is 100% and 0km is 0%
```

```

}
if("rain" in responseJSON):
    returnObject["rain"] = [responseJSON["rain"][key] for key in
responseJSON["rain"]]
return(returnObject)

```

2. brain.py

This file is a utility function that abstracts all unnecessary details and only returns the information that is necessary to be displayed on the hardware side. The logic for the code flow is carried out here.

Python code

```

import weather
from datetime import datetime as dt

def processConditions(myLocation,APIKEY,localityInfo):
    weatherData = weather.get(myLocation,APIKEY)

    finalSpeed = localityInfo["usualSpeedLimit"] if "rain" not in weatherData else
localityInfo["usualSpeedLimit"]/2
    finalSpeed = finalSpeed if weatherData["visibility"]>35 else finalSpeed/2

    if(localityInfo["hospitalsNearby"]):
        # hospital zone
        doNotHonk = True
    else:
        if(localityInfo["schools"]["schoolZone"]==False):
            # neither school nor hospital zone
            doNotHonk = False

```

```

else:
    # school zone
    now = [dt.now().hour,dt.now().minute]
    activeTime = [list(map(int,_.split(":"))) for _ in
localityInfo["schools"]["activeTime"]]
    doNotHonk = activeTime[0][0]<=now[0]<=activeTime[1][0] and
activeTime[0][1]<=now[1]<=activeTime[1][1]

    return({
        "speed" : finalSpeed,
        "doNotHonk" : doNotHonk
    })

```

3. Main.py

The code that runs in a forever loop in the microcontroller. This calls all the utilfunctions from other python files and based on the return value transduces changes in the output hardware display.

Python code

```
import brain
```

```
myLocation = "Chennai,IN"
```

```
APIKEY = "c76d51c15c0e7c6c5f2002ad65efcec1"
```

```
localityInfo = {
```

```
    "schools" : {
```

```
        "schoolZone" : True,
```

```
        "activeTime" : ["7:00","17:30"] # schools active from 7 AM till 5:30 PM
```

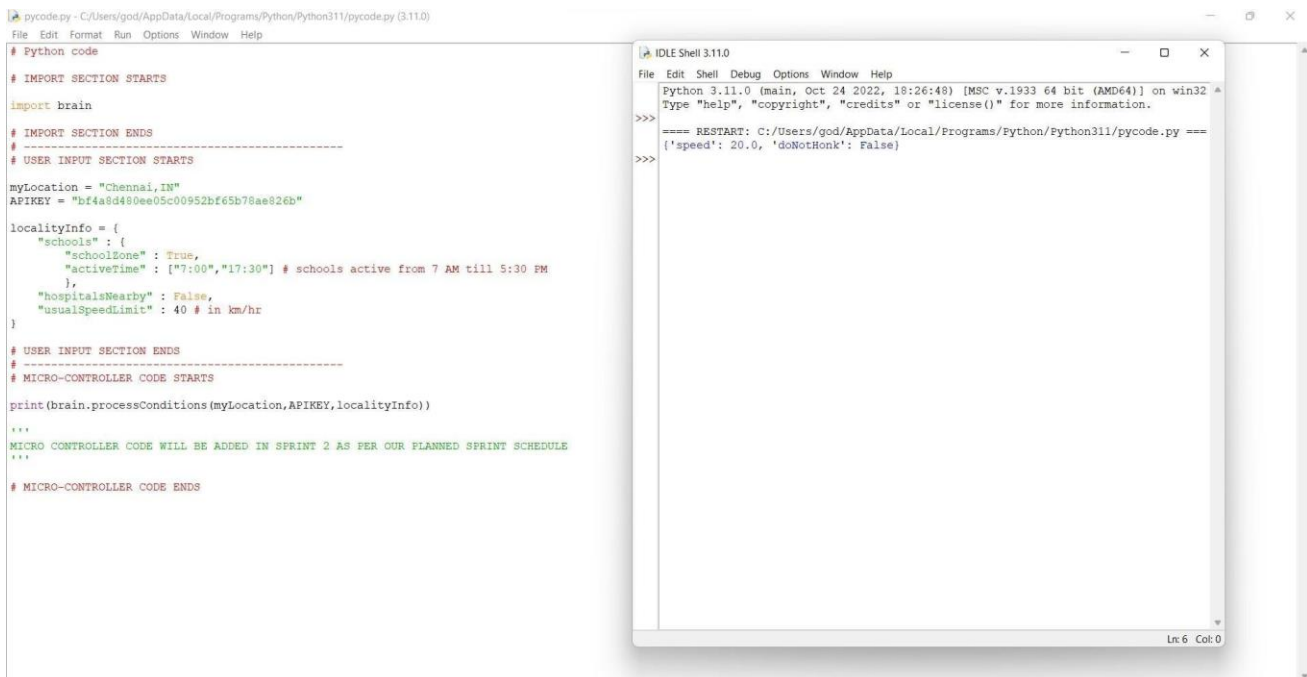
```
    },
```

```
"hospitalsNearby" : False,  
"usualSpeedLimit" : 40 # in km/hr  
}  
  
print(brain.processConditions(myLocation,APIKEY,localityInfo))
```

OUTPUT:

Code Output

```
{'speed': 40, 'doNotHonk': False}
```



The screenshot displays a Python IDE with two windows. The left window, titled 'pcode.py - C:/Users/god/AppData/Local/Programs/Python/Python311/pcode.py (3.11.0)', contains the following Python code:

```
# Python code  
  
# IMPORT SECTION STARTS  
import brain  
  
# IMPORT SECTION ENDS  
# -----  
# USER INPUT SECTION STARTS  
  
myLocation = "Chennai,IN"  
APIKEY = "bf4a8d480ee05c0952bf65b78ae826b"  
  
localityInfo = {  
    "schools" : {  
        "schoolDone" : True,  
        "activeTime" : ["7:00","17:30"] # schools active from 7 AM till 5:30 PM  
    },  
    "hospitalsNearby" : False,  
    "usualSpeedLimit" : 40 # in km/hr  
}  
  
# USER INPUT SECTION ENDS  
# -----  
# MICRO-CONTROLLER CODE STARTS  
  
print(brain.processConditions(myLocation,APIKEY,localityInfo))  
  
...  
MICRO CONTROLLER CODE WILL BE ADDED IN SPRINT 2 AS PER OUR PLANNED SPRINT SCHEDULE  
...  
  
# MICRO-CONTROLLER CODE ENDS
```

The right window, titled 'IDLE Shell 3.11.0', shows the output of the code execution:

```
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>  
==== RESTART: C:/Users/god/AppData/Local/Programs/Python/Python311/pcode.py ====  
{'speed': 20.0, 'doNotHonk': False}  
>>>
```

