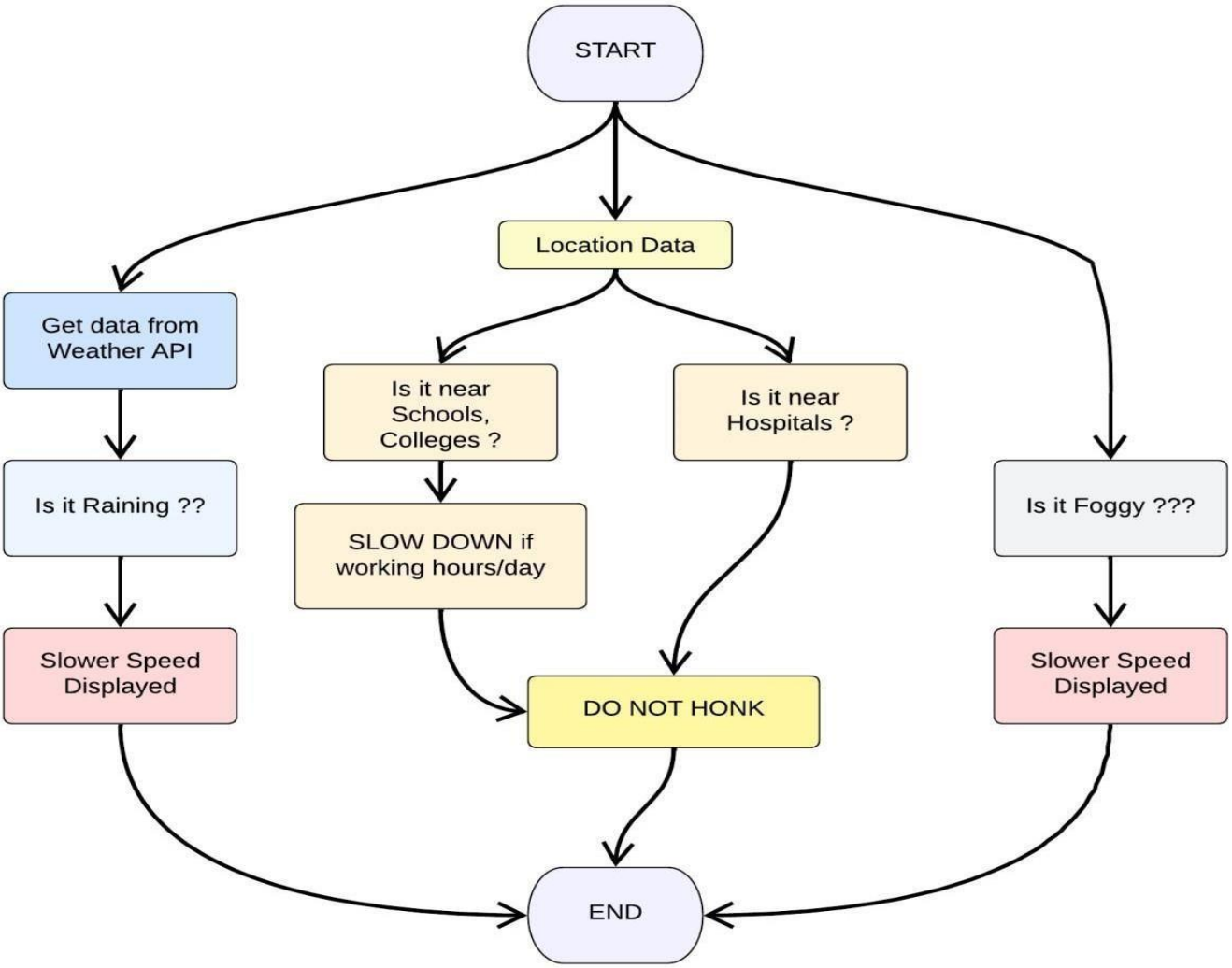
SPRINT 1

Team ID	PNT2022TMID12332
Project Name	Signs with smart connectivity for Better road safety

Code Flow :



brain.py

```
# IMPORT SECTION STARTS
import weather
from datetime import datetime as dt
# IMPORT SECTION ENDS
# UTILITY LOGIC SECTION STARTS
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"]):
 # hzone of the hospital
 doNotHonk = True
 else:
   if(localityInfo["schools"]["schoolZone"]==False):
 # neither hospital zone nor school
    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
 })
```

File Edit Format Run Options Window Help

```
#Python code
# IMPORT SECTION STARTS
import weather
from datetime import datetime as dt
# IMPORT SECTION ENDS
# UTILITY LOGIC SECTION STARTS
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"]):
 # hzone of the hospital
   doNotHonk = True
 else:
   if(localityInfo["schools"]["schoolZone"]==False):
 # neither hospital zone nor school
    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
 })
```

weather.py

```
import requests as reqs
def get(myLocation,APIKEY):
    apiURL =
"https://api.openweathermap.org/data/2.5/weather?q={myLocation}&appid={APIKEY}"
    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,
    }
    if("rain" in responseJSON):
        returnObject["rain"] = [responseJSON["rain"][key] for key in responseJSON["rain"]]
    return(returnObject)
```

File Edit Format Run Options Window Help

```
import requests as reqs

def get(myLocation,APIKEY):
    apiURL = "https://api.openweathermap.org/data/2.5/weather?q={myLocation}&appid={APIKEY}"
    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,
    }
    if("rain" in responseJSON):
        returnObject["rain"] = [responseJSON["rain"][key] for key in responseJSON["rain"]]
    return(returnObject)
```

main.py

```
import brain
# IMPORT SECTION ENDS
# USER INPUT SECTION STARTS
myLocation = "Chennai,IN"
APIKEY = "c7388b7d0d823ee0ee0be65c6fd40411"
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
}
# USER INPUT SECTION ENDS
# MICRO-CONTROLLER CODE STARTS
while True :
   print(brain.processConditions(myLocation,APIKEY,localityInfo))
```

File Edit Format Run Options Window Help

```
import brain
# IMPORT SECTION ENDS
# USER INPUT SECTION STARTS
myLocation = "Chennai,IN"
APIKEY = "c7388b7d0d823ee0ee0be65c6fd40411"
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
}
# USER INPUT SECTION ENDS
# MICRO-CONTROLLER CODE STARTS
while True :
    print(brain.processConditions(myLocation,APIKEY,localityInfo))
```

Output :

Code Output
{'speed': 40, 'doNotHonk': False}