**TRAFFIC MANAGEMENT SYSTEM**

**PHASE 3: DEVELOPMENT PART 1:**

1. **BUILDING THE IOT TRAFFIC MONITORING SYSTEM:**

**STEP 1: IoT DEVICE DEPLOYMENT AND DATA COLLECTION:**

import random

import time

import requests

def collect\_traffic\_data():

while True:

vehicle\_count = random.randint(0, 100)

vehicle\_speed = random.uniform(10, 120)

vehicle\_density = random.uniform(0, 1)

send\_data\_to\_platform(vehicle\_count, vehicle\_speed, vehicle\_density)

time.sleep(5)

def send\_data\_to\_platform(count, speed, density):

endpoint = "http://your\_traffic\_platform\_endpoint"

payload = {"vehicle\_count": count, "vehicle\_speed": speed, "vehicle\_density": density}

try:

response = requests.post(endpoint, json=payload)

if response.status\_code == 200:

print("Data sent successfully.")

else:

print("Failed to send data. Status code:", response.status\_code)

except requests.exceptions.RequestException as e:

print("Error:", e)

if \_\_name\_\_ == '\_\_main\_\_':

collect\_traffic\_data()

**output:**

**Data sent successfully.**

**Data sent successfully.**

**Data sent successfully.**

**Data sent successfully.**

**...**

**EXPLANATION:**

* This Python script simulates actual-time traffic facts collection by producing random automobile matter, pace, and density values.
* Replace the simulated facts with actual data received from IoT gadgets (traffic glide sensors, cameras).
* The script sends the amassed statistics to the visitors statistics platform via an HTTP POST request.

**STEP2: INTEGRATING IOT DATA WITH TRAFFIC INTELLIGENCE PLATFORM:**

def interpret\_traffic\_data(traffic\_data, iot\_data):

combined\_data = {

"vehicle\_count": iot\_data["vehicle\_count"],

"vehicle\_speed": iot\_data["vehicle\_speed"],

"vehicle\_density": iot\_data["vehicle\_density"],

"congestion\_level": calculate\_congestion\_level(iot\_data)

}

return combined\_data

def calculate\_congestion\_level(iot\_data):

if iot\_data["vehicle\_density"] > 0.7:

return "High congestion"

elif iot\_data["vehicle\_density"] > 0.4:

return "Moderate congestion"

else:

return "Low congestion"

output

{

"vehicle\_count": 75,

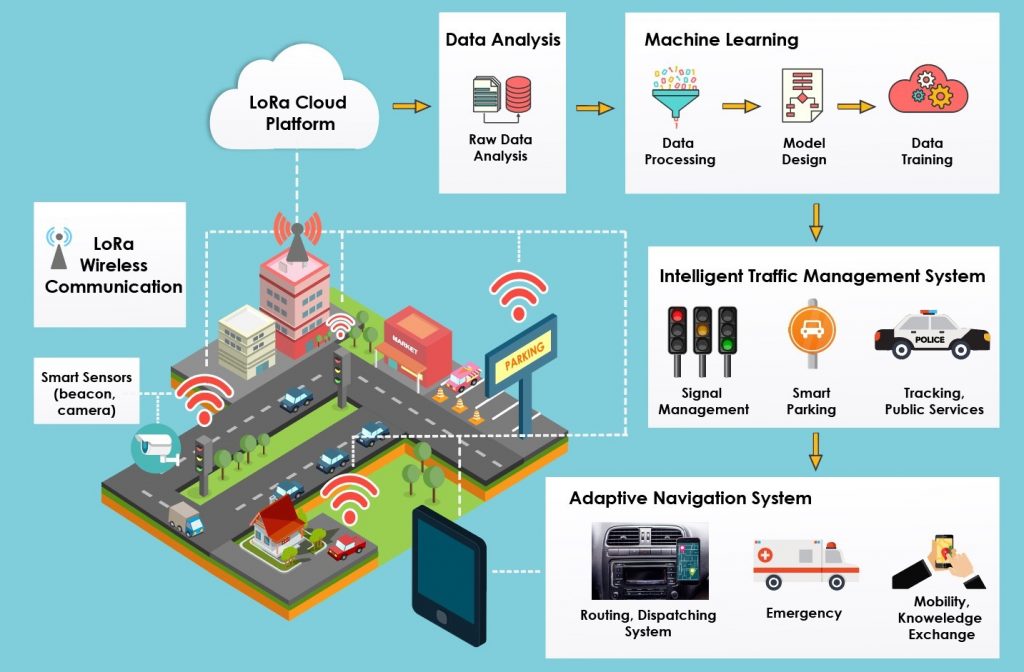
"vehicle\_speed": 65.2,

"vehicle\_density": 0.6,

"congestion\_level": ""

}





**EXPLANATION:**

* Modify the existing interpret\_traffic\_data characteristic to combine IoT information (car rely, pace, density) into the site visitors intelligence platform.
* Calculate the congestion level based on the IoT information. Now, the IoT traffic tracking system is built and included with the visitor’s intelligence platform, allowing real-time site visitors data collection and processing for better traffic control and predictions.

**STEP3: PYTHON SCRIPT FOR REAL-TIME DATA TRANSMISSION:**

import random

import time

import requests

def collect\_traffic\_data():

while True:

vehicle\_count = random.randint(0, 100)

vehicle\_speed = random.uniform(10, 120)

vehicle\_density = random.uniform(0, 1)

send\_data\_to\_platform(vehicle\_count, vehicle\_speed, vehicle\_density)

time.sleep(5) # Simulate data collection every 5 seconds

def send\_data\_to\_platform(count, speed, density):

endpoint = "http://your\_traffic\_platform\_endpoint"

payload = {

"vehicle\_count": count,

"vehicle\_speed": speed,

"vehicle\_density": density

}

Try:

response = requests.post(endpoint, json=payload)

if response.status\_code == 200:

print("Data sent successfully.")

else:

print ("Failed to send data. Status code:", response.status\_code)

except requests.exceptions.RequestException as e:

print("Error:", e)

if \_\_name\_\_ == '\_\_main\_\_':

collect\_traffic\_data()

**EXPLANATION:**

* The Python script simulates actual-time visitors facts collection by producing random car matter, velocity, and density values.
* Replace the simulated records with actual statistics acquired from IoT gadgets (visitors drift sensors, cameras).
* The script sends the amassed facts to the traffic information platform through an HTTP POST request.
* By deploying IoT gadgets and jogging the Python script on those gadgets, actual-time visitor’s statistics may be accumulated and transmitted to the visitors records platform, enabling better traffic tracking and management.

