Traffic Management

Introduction

Over the years, there has been a sudden increase in the number of vehicles on the road. Traffic congestion is a growing problem everyone faces in their daily life. Manual control of trace by traffic police has not proved to be efficient. Also, the predefined set time for the signal at all circumstances (low and high traffic density) has not solved this problem. A model to effectively solve the above-mentioned problems by using Internet of Things (IoT) is proposed. Smart Traffic management provide an organized, integrated approach to minimize the congestion and   improving safety on city streets through connected technology.

Source Code

Import time

Import random

Class Vehicle:

def \_init\_(self, vehicle\_id, speed):

self.vehicle\_id = vehicle\_id

self.speed = speed

class TrafficLight:

def \_init\_(self):

self. state = 'red'

def switch state(self):

self. state = 'green' if self. state == 'red' else 'red'

class TrafficManager:

def \_init\_(self, num\_vehicles, num\_iterations):

self. vehicles = [Vehicle(i, random.randint(30, 70))

for i in range(num\_vehicles)]

self.traffic\_light = TrafficLight ()

self.num\_iterations = num\_iterations

def run\_simulation(self):

for iteration in range(self.num\_iterations):

print(f"Iteration {iteration + 1}: Traffic Light is {self.traffic\_light.state}")

for vehicle in self. vehicles:

if self.traffic\_light.state == 'green':

distance = vehicle.speed \* 2

print(f"Vehicle{vehicle.vehicle\_id}moves{distance}meters.")

else:

print(f"Vehicle {vehicle.vehicle\_id} stops at the red light.")

self.traffic\_light.switch\_state()

time.sleep(1)

def main():

num\_vehicles = 5

num\_iterations = 5

traffic\_manager = TrafficManager (num\_vehicles, num\_iterations)

traffic\_manager.run\_simulation ()

If \_name\_ == "\_main\_":

main ()

Output

Iteration 1: Traffic Light is green

Vehicle 0 moves 120 meters.

Vehicle 1 moves 80 meters.

Vehicle 2 moves 160 meters.

Vehicle 3 moves 60 meters.

Vehicle 4 moves 100 meters.

Iteration 2: Traffic Light is red

Vehicle 0 stops at the red light.

Vehicle 1 stops at the red light.

Vehicle 2 stops at the red light.

Vehicle 3 stops at the red light.

Vehicle 4 stops at the red light.

Iteration 3: Traffic Light is green

Vehicle 0 moves 120 meters.

Vehicle 1 moves 80 meters.

Vehicle 2 moves 160 meters.

Vehicle 3 moves 60 meters.

Vehicle 4 moves 100 meters.

Iteration 4: Traffic Light is red

Vehicle 0 stops at the red light.

Vehicle 1 stops at the red light.

Vehicle 2 stops at the red light.

Vehicle 3 stops at the red light.

Vehicle 4 stops at the red light.

Iteration 5: Traffic Light is green

Vehicle 0 moves 120 meters.

Vehicle 1 moves 80 meters.

Vehicle 2 moves 160 meters.

Vehicle 3 moves 60 meters.

Vehicle 4 moves 100 meters.

Explanation of the output:

This code simulates a traffic system with 5 vehicles and 5 iterations. Each iteration represents a change in the traffic light state. When the traffic light is green, all vehicles move twice their speed. When the traffic light is red, all vehicles stop at the intersection. The simulation is run for 5 iterations, displaying the traffic light state and vehicle movement in each iteration.</s>

Please note that the actual output may vary due to the random nature of vehicle speeds and the traffic light switching time.</s> </s> Please also note that the output of the simulation includes a delay of 1 second after each iteration due to the time.sleep(1) function