* SMART TRAFFIC MANAGEMENT SYSTEM

Creating a complete smart traffic management system with IoT involves multiple components and is quite complex. Here, I can provide you with a simplified example of an IoT-based traffic monitoring system using Python and the Raspberry Pi. This example assumes you have basic knowledge of Python, Raspberry Pi, and access to IoT hardware like sensors and cameras.

Program code

```python

import time

import RPi.GPIO as GPIO

from picamera import PiCamera

from datetime import datetime

# Set up GPIO pin for a motion sensor

GPIO.setmode(GPIO.BOARD)

PIR\_PIN = 7

GPIO.setup(PIR\_PIN, GPIO.IN)

# Initialize the PiCamera

camera = PiCamera()

def capture\_image():

timestamp = datetime.now().strftime("%Y-%m-%d\_%H-%M-%S")

image\_filename = f"image\_{timestamp}.jpg"

camera.capture(image\_filename)

return image\_filename

try:

while True:

if GPIO.input(PIR\_PIN):

print("Motion detected! Capturing image...")

image\_file = capture\_image()

print(f"Image captured: {image\_file}")

time.sleep(1)

except KeyboardInterrupt:

print("Traffic monitoring stopped.")

GPIO.cleanup()

```

This code uses a motion sensor connected to the Raspberry Pi to detect motion (e.g., a vehicle passing by). When motion is detected, it captures an image using the PiCamera module. You can expand upon this basic example by integrating other sensors, communication protocols (e.g., MQTT for sending data to a server), and a web interface for monitoring and controlling the traffic system.

Remember that building a comprehensive smart traffic management system involves more components, such as traffic lights, communication with a central server, data analysis, and decision-making algorithms. You may need to break down your project into smaller, manageable parts and develop them incrementally.